

6 Time Management

Time management is concerned with the mechanisms for controlling the advancement of federates along the federation time axis during the execution. In general, time advances must be coordinated with object management services so that information is delivered to federates (e.g., state updates and interactions) in a timely and ordered fashion, thereby enabling federates to satisfy requirements for reproducing causal behavior in the system being modeled. For example, certain federates may require that notifications of events be delivered to the federate in time stamp order, and are not delivered in the federate's past (time stamp less than the federate's current time). The time management services are intended to support federations that include federates with different ordering and delivery requirements. For further information, see the "HLA Time Management: Design Document" [HLA TM].

6.1 Request Federation Time

Federate Initiated

Requests a current estimate of the federation time. Federation time is defined to be the minimum of Lower Bound Time Stamp (LBTS) and the current value of the federate's Logical Time (LT).

Supplied Parameters

None

Returned Parameters

The current minimum of LBTS and LT

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The federate receives the current value of federation time

Exceptions

Federate is not a federation execution member

RTI internal error

Related Services

Request LBTS

Request Federate Time

Request Minimum Next Event Time

6.2 Request LBTS

Federate Initiated

Requests the current value of LBTS.

Supplied Parameters

None

Returned Parameters

The current value of LBTS

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The federate receives the current value of LBTS

Exceptions

Federate is not a federation execution member

RTI internal error

Related Services

Request Federation Time

Request Federate Time

Request Minimum Next Event Time

6.3 Request Federate Time

Federate Initiated

Requests the current value of the federate's Logical Time (LT).

Supplied Parameters

None

Returned Parameters

The current value of LT.

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The federate receives the current value of LT

Exceptions

Federate is not a federation execution member

RTI internal error

Related Services

Request Federation Time

Request LBTS

Request Minimum Next Event Time

6.4 Request Minimum Next Event Time

Federate Initiated

Requests the minimum of LBTS and the time stamp of the next Time Stamp Ordered (TSO) message that is held by the RTI for delivery to the requesting federate.

Supplied Parameters

None

Returned Parameters

The minimum of the following:

LBTS and

the time of the message at the head of the TSO queue (if it contains a message).

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The federate receives the minimum next event time.

Exceptions

Federate is not a federation execution member

RTI internal error

Related Services

Request Federation Time

Request LBTS

Request Federate Time

6.5 Set Lookahead

Federate Initiated

Sets the desired value of the federate's lookahead.

Supplied Parameters

A desired value of lookahead

Returned Parameters

None

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The RTI is informed of the federate's new value of lookahead

Exceptions

Invalid Lookahead Time

Federate is not a federation execution member

RTI internal error

Related Services

Request Lookahead

6.6 Request Lookahead

Federate Initiated

Queries the RTI for the current value of the federate's lookahead. The current value of lookahead may differ temporarily from the desired lookahead given in the *Set Lookahead* service if the value of lookahead has been reduced.

Supplied Parameters

None

Returned Parameters

The federate's current value of lookahead

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The federate receives the current value of it's lookahead

Exceptions

Federate is not a federation execution member

RTI internal error

Related Services

Set Lookahead

6.7 Time Advance Request

Federate Initiated

The *Time Advance Request* service requests an advance of the federate's logical time. Invocation of this service implies that the following messages are eligible for delivery to the federate: (i) all incoming receive ordered messages, and (ii) all messages using other ordering services with time stamp less than or equal to the specified time. After invoking *Time Advance Request*, the messages are passed to the federate by the RTI invoking the *Discover Object*, *Remove Object*, *Receive Interaction* and *Reflect Attribute Values* services. By invoking *Time Advance Request* with the specified time, the federate is guaranteeing that it will not generate a Time Stamp Ordered (TSO) message at any time in the future with time stamp less than the specified time, plus that federate's current lookahead. A *Time Advance Grant* completes this request and indicates to the federate that it has advanced its logical time to the specified time.

Supplied Parameters

A value of federation time

Returned Parameters

None

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The RTI is informed of the federate's desire to advance time

Exceptions

Invalid federation time

Time Advance Request or *Next Event Request*, or *Flush Queue Request* already pending

Federate is not a federation execution member

RTI internal error

Related Services

Next Event Request

Flush Queue Request

Time Advance Grant

6.8 Next Event Request

Federate Initiated

The *Next Event Request* service requests the next TSO message from the RTI, provided that message has a time stamp no greater than the logical time specified in the request. The invocation of this service implies that the following messages are eligible for delivery to the federate:

- All receive ordered messages,
- The smallest time stamped TSO message that will ever be delivered in the future with time stamp less than or equal to the specified time, and all other TSO messages containing this same time stamp value, and
- All messages using other ordering services with time stamp less than or equal to the specified time.

These messages are delivered to the federate by the RTI calling the *Discover Object*, *Remove Object*, *Receive Interaction* or *Reflect Attribute Values* services provided by the federate. A *Time Advance Grant* completes this request and indicates to the federate that it has advanced its logical time to the latest time stamp of all the TSO messages that are delivered, if any, or to the specified time if no TSO messages were delivered.

Supplied Parameters

A value of federation time

Returned Parameters

None

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The RTI is informed of the federate's desire to advance time

Exceptions

Invalid federation time

Time Advance Request or *Next Event Request*, or *Flush Queue Request* already pending

Federate is not a federation execution member

RTI internal error

Related Services

Time Advance Request

Flush Queue Request

Time Advance Grant

6.9 Flush Queue Request

Federate Initiated

The *Flush Queue Request* releases all messages stored in the RTI's internal queues and delivers them to the federate invoking this service. Time Stamp Ordered (TSO) messages are delivered, despite the fact that the RTI may not be able to guarantee future messages containing a smaller time stamp could arrive. If the federate does not receive any additional TSO messages with time stamp less than the specified time, then the federate's LT may be advanced up to the specified time. Received messages are delivered to the federate by the RTI calling the *Discover Object*, *Remove Object*, *Receive Interaction* or *Reflect Attribute Values* services provided by the federate. A *Time Advance Grant* completes this request and indicates to the federate that it has advanced its logical time to the minimum of the specified time, LBTS, and the time stamp of the next TSO message to be received.

Supplied Parameters

A value of federation time

Returned Parameters

None

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Post-conditions

The RTI is informed of the federate's desire to advance time

Exceptions

Invalid federation time

Time Advance Request or *Next Event Request*, or *Flush Queue Request* already pending

Federate is not a federation execution member

RTI internal error

Related Services

Time Advance Request

Next Event Request

Time Advance Grant

6.10 Time Advance Grant †

RTI Initiated

Invocation of this service indicates a prior request to advance Logical Time (LT) has been honored. The parameter of this service indicates that LT for the federate has been advanced to this value and no additional Time Stamped Ordered (TSO) messages will be delivered in the future with a time stamp less than this value.

Supplied Parameters

A value of federation time

Returned Parameters

None

Pre-conditions

The federation execution exists

The federate has joined that federation execution

Any of the *Time Advance Request*, *Next Event Request* or *Flush Queue Request* services were previously invoked

Post-conditions

No additional TSO messages will be received with a time stamp less than the provided time

Exceptions

Invalid federation time

Time Advance Request, *Next Event Request* or *Flush Queue Request* was not pending

Federate internal error

Related Services

Time Advance Request

Next Event Request

Flush Queue Request

7 Data Distribution Management

To support efficient data distribution across a federation, the RTI provides a set of services which facilitate the explicit management of data distribution. The fundamental concept to support data distribution is called *routing spaces*. A *routing space* is a multidimensional coordinate system in which federates express an interest for either receiving data or sending data.

To use routing spaces, each federation defines the allowable routing spaces for the federation execution, including the dimensions (variables) of the routing space. Routing spaces are then initialized in the FED with a name and the number of dimensions.

Using declaration management services (Section 3), federates specify by class and attribute designator and by interaction class, the *types* of data they will send or receive. Routing spaces are used by federates to specify the distribution conditions for the *specific* data they are sending or expect to receive. Each federate decides which of the federation routing spaces are useful to them and defines the portions (subsets) of those routing spaces that specify (from the federate's perspective) *regions*, or logical areas of interest particular to the federate, by putting bounds (*extents*) on the dimensions of the selected routing space. The federate then uses these regions to both specify conditions (*Create Subscription Region*) under which they expect to receive the object state data and interactions they specified using declaration management services (*Subscribe Object Class Attribute* and *Subscribe Interaction Class*) and to specify conditions under which they are providing data (*Create Update Region and Associate Update Region*).

Specifying a subscription region, the federate tells the RTI to only deliver data which fall within the bounds (extents) of the region specified by that federate. Specifying an update region and associating that update region with a particular object instance, means that the federate will ensure that the characteristics of the object instance or interaction which map to the dimensions of the routing space fall within the bounds (extents) of the associated region at the time that the attribute update or send interaction call is issued. This implies that the federate is monitoring these added characteristics for each of the attributes owned by the federate. As the state of the objects change, the federate may need to either adjust the bounds on the associated regions (*Modify Region*) or change the association to another region (*Associate Update Region*).

The routing space, regions, and association data is used by the RTI to distribute data. When an update region and subscription regions of different federates overlap, the RTI ensures that the attribute updates and interactions associated with that update region are routed to federates with subscription regions which overlap the sender's update region.

It is important to note that the dimensions of routing spaces need not necessarily map to attributes in the FED. The data distribution management services provide a federation the capability to specify data distribution on characteristics of objects other than those exchanged as part of federation execution. By specifying routing spaces and regions using attributes specified in the FED, the data distribution management services provide a mechanism to control data distribution based on values of attributes.

Each federate can create multiple update and subscription regions. Update regions may be associated with individual objects that have been registered with the RTI. As an example, a federate might have a subscription region for each sensor system being simulated.

Figure 5 shows one update region (U1) and two subscription regions (S1, S2) within a two dimensional routing space. In this example, U1 and S1 overlap and therefore attributes and interactions associated with U1 will be routed by the RTI to the federate that created S1. In contrast U1 and S2 do not overlap and attributes and interactions will not be routed from U1 to the federate that created S2.

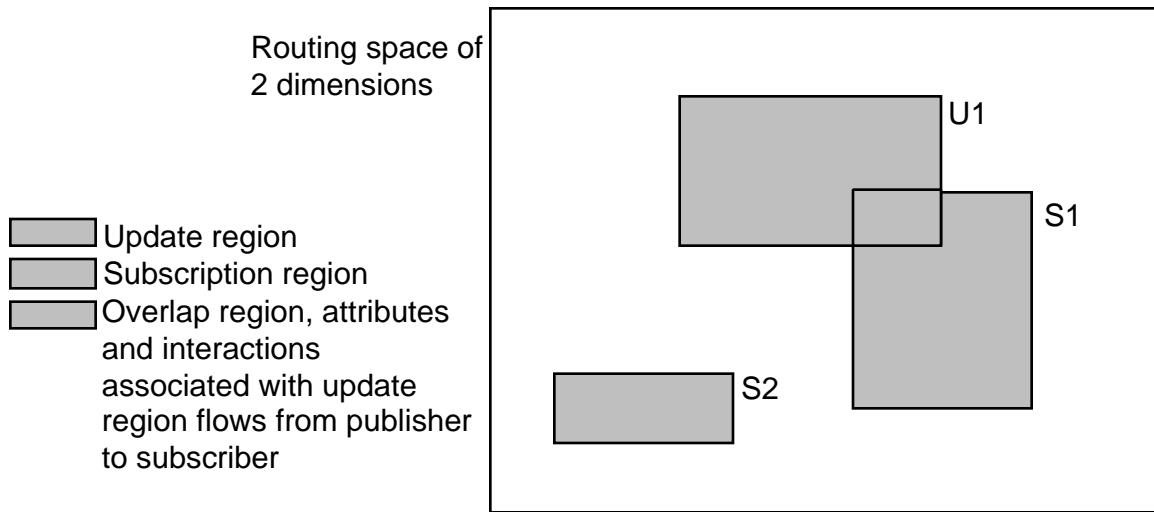


Figure 5. Routing Space Example

Disclaimer: There is ongoing exploration of the Data Distribution Management services, especially with respect to the interaction of Time Management and Data Distribution Management. As such, this specification contains no information concerning aspects of the interaction, such as at what point in federation time a certain Data Distribution Management service takes effect.

7.1 Create Update Region

Federate Initiated

Creates an update region that is a subset of the specified routing space. The extent set bounds the region within the routing space.

Supplied Parameters

A routing space designator

A set of extents

Returned Parameters

An update region designator

Pre-conditions

The routing space is defined in the FED

Post-conditions

An update region exists that is a subset of the specified routing space

Exceptions

Routing space not defined in the FED

Invalid extents

Federate is not an execution member

RTI internal error

Related Services

Create Subscription Region

Associate Update Region

Change Thresholds †

Modify Region

Delete Region

7.2 Create Subscription Region

Federate Initiated

Creates a subscription region that is a subset of the specified routing space. The extent set bounds the region within the routing space. In creating a subscription region, the federate tells the RTI to only deliver data which fall within the bounds (extents) of the region.

Supplied Parameters

A routing space designator

A set of extents

Returned Parameters

A subscription region designator

Pre-conditions

The routing space is defined in the FED

Post-conditions

A subscription region exists that is a subset of the specified routing space

Exceptions

Routing space not defined in the FED

Invalid extents

Federate is not a federation execution member

RTI internal error

Related Services

Create Update Region

Associate Update Region

Change Thresholds †

Modify Region

Delete Region

Subscribe Object Class Attribute

Subscribe Interaction

7.3 Associate Update Region

Federate Initiated

This service associates an update region with

- attributes of a specific object, or
- an interaction class.

Associating an update region with a particular object instance or interaction class, means that the federate will ensure that the characteristics of the object instance or interaction which map to the dimensions of the routing space fall within the bounds (extents) of the associated region at the time that the attribute update or send interaction call is issued. This implies that the federate is monitoring these added characteristics for each of the attributes owned by the federate. As the state of the objects change, as required the federate must either adjust the bounds on the associated regions (*Modify Region*) or change the association to another region (*Associate Update Region*).

The association is used by the *Update Attribute Values* and *Send Interaction* services to route data to subscribers whose subscription regions overlap the specified update region.

Supplied Parameters

A update region designator

For interaction classes:

An interaction class

or for objects:

An object ID

A set of attributes

An indication to form or break the association

Returned Parameters

None

Pre-conditions

The update region exists

The object ID is registered

The federate is publishing the specified attributes, or

The federate is publishing the specified interaction class

Post-conditions

The update region is associated with the object ID or interaction class for future *Update Attribute Values* or *Send Interactions* service invocations

Exceptions

Region not known

Object not known

Attributes not defined in the FED

Interaction class not defined in the FED

Federate is not a federation execution member

RTI internal error

Related Services

Create Update Region

Change Thresholds †

Modify Region

Delete Region

Update Attribute Values

Send Interaction

7.4 Change Thresholds †

RTI Initiated

Change Thresholds provides the federate with the new values of the thresholds for each of the dimensions of a routing space. The new threshold data should be used by the federate to decide when to invoke the *Modify Region* service.

Supplied Parameters

An update or subscription region designator

A set of thresholds

Returned Parameters

None

Pre-conditions

The region exists

Post-conditions

The federate is informed of the new threshold values for the region

Exceptions

Region not known

Federate internal error

Related Services

Create Subscription Region

Create Update Region

Modify Region

7.5 Modify Region

Federate Initiated

Modify Region changes the bounds of the update or subscription region to reflect the specified set of extents.

Supplied Parameters

An update or subscription region designator

A set of extents

Returned Parameters

None

Pre-conditions

The region exists

Post-conditions

The region has a new extent

Exceptions

Region not known

Invalid extents

Federate is not a federation execution member

RTI internal error

Related Services

Create Subscription Region

Create Update Region

Change Thresholds †

7.6 Delete Region

Federate Initiated

Deletes the specified update or subscription region.

Supplied Parameters

An update or subscription region designator

Returned Parameters

None

Pre-conditions

The region exists

Post-conditions

The region no longer exists

The region is no longer associated with any interaction class or object attributes

Exceptions

Region not known

Federate is not a federation execution member

RTI internal error

Related Services

Create Subscription Region

Create Update Region

8 HLA IDL Application Programmer's Interface

```
// File: baseTypes.idl

#ifndef __baseTypes
#define __baseTypes

interface baseTypes {

    typedef unsigned short RTI_UShort;
    typedef short RTI_Short;
    typedef unsigned long RTI_ULong;
    typedef long RTI_Long;
    typedef double RTI_Double;
    typedef float RTI_Float;
    typedef boolean RTI_Boolean;
    typedef any RTI_Datum;
    typedef string RTI_String;
    typedef unsigned long RTI_Id;

};

#endif
```

```
//File: rtiTypes.idl

#ifndef __rtiTypes
#define __rtiTypes

#include "baseTypes.idl"

interface rtiTypes: baseTypes {

    enum OrderType {
        ORDER_TYPE_NONE,
        RECEIVE,
        TIMESTAMP
    };

    enum OwnershipDivestitureCondition {
        NEGOTIATED,
        UNCONDITIONAL
    };

    enum ObjectRemovalReason {
        OBJECT_OUT_OF_REGION,
        OBJECT_DELETED,
        NO_LONGER_SUBSCRIBED
    };

    enum TransportType {
        TRANSPORT_TYPE_NONE,
        RELIABLE,
        BEST EFFORT
    };

    enum ResignAction {
        RELEASE_ATTRIBUTES,
        DELETE_OBJECTS,
        DELETE_OBJECTS_AND_RELEASE_ATTRIBUTES,
        NO_ACTION
    };

    typedef RTI_ULong UniqueId;

    typedef RTI_Double FederationTimeDelta;
    typedef RTI_Double FederationTime;
    typedef RTI_Double WallClockTime;
    typedef RTI_Double TickTime;

    typedef RTI_UShort FederateHandle;
    typedef RTI_String FederateName;
    typedef sequence <FederateHandle> FederateHandleSet;

    typedef RTI_UShort ObjectClassHandle;
    typedef RTI_String ObjectClassName;

    typedef RTI_UShort InteractionClassHandle;
    typedef RTI_String InteractionClassName;
```

```
typedef RTI_UShort AttributeHandle;
typedef RTI_StringAttributeName;
typedef RTI_DatumAttributeValue;
typedef sequence <AttributeHandle> AttributeHandleSet;
typedef struct AttributeHandleValuePair_s {
    AttributeHandle name;
    AttributeValue value;
} AttributeHandleValuePair;
typedef sequence <AttributeHandleValuePair> AttributeHandleValuePairSet;

typedef RTI_UShort ParameterHandle;
typedef RTI_String ParameterName;
typedef RTI_Datum ParameterValue;
typedef struct ParameterHandleValuePair_s {
    ParameterHandle name;
    ParameterValue value;
} ParameterHandleValuePair;
typedef sequence <ParameterHandleValuePair> ParameterHandleValuePairSet;

typedef RTI_ULong ObjectId;
typedef sequence <ObjectId> ObjectIdSet;
typedef RTI_UShort ObjectIdCount;

typedef struct EventRetractionHandle_s {
    FederationTime theEventTime;
    UniqueId theSerialNumber;
    FederateHandle sendingFederate;
} EventRetractionHandle;

typedef RTI_Long SpaceHandle;
typedef RTI_String SpaceName;

typedef RTI_String FederationExecutionName;

typedef RTI_String FileName;

typedef RTI_String PauseLabel;

typedef RTI_String SaveLabel;

typedef RTI_Datum UserSuppliedTag;

typedef struct Range_s {
    RTI_ULong first;
    RTI_ULong last;
} Range;
typedef sequence <Range> Extents;
typedef sequence <Extents> ExtentSet;

typedef RTI_ULong Region;

typedef struct Threshold_s {
    RTI_ULong value;
} Threshold;
```

```
typedef sequence <Threshold> ThresholdSet;

exception AttributeAlreadyOwned {};
exception AttributeNotDefined {};
exception AttributeNotKnown {};
exception AttributeNotOwned {};
exception AttributeNotPublished {};
exception AttributeNotSubscribed {};
exception CouldNotDiscover {};
exception CouleNotLocateFED{};
exception CouldNotInitiateRestore{};
exception DeletePrivilegeNotHeld {};
exception InvalidFED{};
exception EventNotKnown {};
exception FederateAlreadyPaused {};
exception FederateAlreadyExecutionMember {};
exception FederateDoesNotExist {};
exception FederateInternalError {};
exception FederateNotExecutionMember {};
exception FederateNotPaused {};
exception FederateNotPublishingInteractionClass {};
exception FederateOwnsAttributes {};
exception FederatesCurrentlyJoined {};
exception FederationAlreadyPaused {};
exception FederationExecutionAlreadyExists {};
exception FederationExecutionDoesNotExist {};
exception FederationNotPaused {};
exception FederationTimeAlreadyPassed {};
exception IdSupplyExhausted {};
exception InteractionClassNotDefined {};
exception InteractionClassNotKnown {};
exception InteractionClassNotPublished {};
exception InteractionParameterNotDefined {};
exception InteractionParameterNotKnown {};
exception InvalidDivestitureCondition {};
exception InvalidExtents {};
exception InvalidFederationTime {};
exception InvalidLookAheadTime {};
exception InvalidObjectId {};
exception InvalidOrderType {};
exception InvalidRetractionHandle {};
exception InvalidResignAction {};
exception InvalidTransportType {};
exception NameNotFound {};
exception NoPauseRequested {};
exception NoResumeRequested {};
exception ObjectClassNotDefined {};
exception ObjectClassNotKnown {};
exception ObjectClassNotPublished {};
exception ObjectClassNotSubscribed {};
exception ObjectNotKnown {};
exception ObjectIdAlreadyLinked {};
exception RegionNotKnown {};
exception RestoreNotRequested {};
exception RTIinternalError {};
```

```
exception SpaceNotDefined {};
exception SaveInProgress {};
exception SaveNotInitiated {};
exception SpecifiedSaveLabelDoesNotExist {};
exception TimeAdvanceAlreadyInProgress {};
exception TimeAdvanceNotInProgress {};
exception TooManyIdsRequested {};
exception UnableToPerformSave {};
exception UnimplementedService {};
exception UnknownLabel {};

};

#endif
```

```
//File: rtiAmbServices.idl

#ifndef __rtiAmbServices
#define __rtiAmbServices

#include "rtiTypes.idl"

///////////////////////////////
// Federation Management Services //
///////////////////////////////

interface rtiAmbassador: rtiTypes {

    // 2.1
    void createFederationExecution (
        in FederationExecutionName      theName,
        in FileName                      FED)
    raises (
        FederationExecutionAlreadyExists,
        CouleNotLocateFED,
        InvalidFED,
        RTIinternalError,
        UnimplementedService);

    // 2.2
    void destroyFederationExecution (
        in FederationExecutionName     executionName)
    raises (
        FederatesCurrentlyJoined,
        FederationExecutionDoesNotExist,
        RTIinternalError,
        UnimplementedService);

    // 2.3
    void joinFederationExecution (
        in FederateName                theFederateName,
        in FederationExecutionName     executionName,
        out FederateHandle             theFederate)
    raises (
        FederateAlreadyExecutionMember,
        FederationExecutionDoesNotExist,
        RTIinternalError,
        UnimplementedService);

    // 2.4
    void resignFederationExecution (
        in ResignAction                 theAction)
    raises (
        FederateOwnsAttributes,
        FederateNotExecutionMember,
        FederationExecutionDoesNotExist,
        InvalidResignAction,
        RTIinternalError,
        UnimplementedService);
```

```
// 2.5
void requestPause (
    in PauseLabel label)
raises (
    FederationAlreadyPaused,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.7
void pauseAchieved (
    in PauseLabel label)
raises (
    UnknownLabel,
    NoPauseRequested,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.8
void requestResume ()
raises (
    FederationNotPaused,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.10
void resumeAchieved ()
raises (
    NoResumeRequested,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.11
void requestFederationSave (
    in SaveLabel          label,
    in FederationTime    theTime)
raises (
    FederationTimeAlreadyPassed,
    InvalidFederationTime,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void requestFederationSave_OVER0 (
    in SaveLabel  label)
raises (
    SaveInProgress,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.13
```

```
void federateSaveBegun ()
raises (
    SaveNotInitiated,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void federateSaveBegun_OVER0 (
    in FederationTime      theTime)
raises (
    SaveNotInitiated,
    InvalidFederationTime,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.14
void federateSaveAchieved ()
raises (
    SaveNotInitiated,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void federateSaveNotAchieved ()
raises (
    SaveNotInitiated,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.15
void requestRestore (
    in SaveLabel      label)
raises (
    SpecifiedSaveLabelDoesNotExist,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 2.17
void restoreAchieved (
    in SaveLabel  label)
raises (
    UnknownLabel,
    RestoreNotRequested,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void restoreNotAchieved (
    in SaveLabel  label)
raises (
    UnknownLabel,
    RestoreNotRequested,
```

```
FederateNotExecutionMember,
RTIinternalError,
UnimplementedService);

///////////////////////////////
// Declaration Management Services //
///////////////////////////////

// 3.1
void publishObjectClass (
    in ObjectClassHandle theClass,
    in AttributeHandleSet attributeList)
raises (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void unpublishObjectClass (
    in ObjectClassHandle theClass)
raises (
    ObjectClassNotDefined,
    FederateOwnsAttributes,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 3.2
void publishInteractionClass (
    in InteractionClassHandle theInteraction)
raises (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void unpublishInteractionClass (
    in InteractionClassHandle theInteraction)
raises (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 3.3
void subscribeObjectClassAttribute (
    in ObjectClassHandle theClass,
    in AttributeHandle theHandle)
raises (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);
```

```
void subscribeObjectClassAttribute_OVER0 (
    in ObjectClassHandle    theClass,
    in AttributeHandle      theAttribute,
    in Region                theRegion)
raises (
    ObjectClassNotDefined,
    AttributeNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void unsubscribeObjectClassAttribute (
    in ObjectClassHandle    theClass)
raises (
    ObjectClassNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void unsubscribeObjectClassAttribute_OVER0 (
    in ObjectClassHandle    theClass,
    in Region                theSubscribeRegion)
raises (
    ObjectClassNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 3.4
void subscribeInteractionClass (
    in InteractionClassHandle    theClass)
raises (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void subscribeInteractionClass_OVER0 (
    in InteractionClassHandle    theClass,
    in Region                theRegion)
raises (
    InteractionClassNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void unsubscribeInteractionClass (
    in InteractionClassHandle    theClass)
raises (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
```

```
RTIinternalError,
UnimplementedService);

void unsubscribeInteractionClass_OVER0 (
    in InteractionClassHandle      theClass,
    in Region                      theRegion)
raises (
    InteractionClassNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

///////////////////////////////
// Object Management Services //
///////////////////////////////

// 4.1
void requestId (
    in ObjectIdCount      theCount,
    out ObjectIdSet        theList)
raises (
    TooManyIdsRequested,
    IdSupplyExhausted,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.2
void registerObject (
    in ObjectClassHandle   theClass,
    in ObjectId            theObject)
raises (
    InvalidObjectId,
    ObjectIdAlreadyLinked,
    ObjectClassNotDefined,
    ObjectClassNotPublished,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.3
void updateAttributeValues (
    in ObjectId                  theObject,
    in AttributeHandleValuePairSet attributeList,
    in FederationTime           theTime,
    in UserSuppliedTag          theTag,
    out EventRetractionHandle   theHandle)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidFederationTime,
    FederateNotExecutionMember,
    RTIinternalError,
```

```
        UnimplementedService);

// 4.6
void sendInteraction (
    in InteractionClassHandle          theInteraction,
    in ParameterHandleValuePairSet     parameters,
    in FederationTime                 theTime,
    in UserSuppliedTag                theTag,
    out EventRetractionHandle         theHandle)
raises (
    InteractionClassNotPublished,
    InteractionClassNotDefined,
    InteractionParameterNotDefined,
    InvalidFederationTime,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.8
void deleteObject (
    in ObjectId                      theObject,
    in FederationTime                 theTime,
    in UserSuppliedTag                theTag,
    out EventRetractionHandle         theHandle)
raises (
    DeletePrivilegeNotHeld,
    ObjectNotKnown,
    InvalidFederationTime,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.10
void changeAttributeTransportType (
    in ObjectId                      theObject,
    in AttributeHandleSet             attributeList,
    in TransportType                  theType)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidTransportType,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.11
void changeAttributeOrderType (
    in ObjectId                      theObject,
    in AttributeHandleSet             attributeList,
    in OrderType                     theType)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
```

```
    InvalidOrderType,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.12
void changeInteractionTransportType (
    in InteractionClassHandle      theClass,
    in TransportType                theType)
raises (
    InteractionClassNotDefined,
    FederateNotPublishingInteractionClass,
    InvalidTransportType,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.13
void changeInteractionOrderType (
    in InteractionClassHandle      theClass,
    in OrderType                   theType)
raises (
    InteractionClassNotDefined,
    FederateNotPublishingInteractionClass,
    InvalidOrderType,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.14
void requestObjectAttributeValuesUpdate (
    in ObjectId                  theObject,
    in AttributeHandleSet attributeList)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void requestClassAttributeValuesUpdate (
    in ObjectClassHandle   theClass,
    in AttributeHandleSet attributeList)
raises (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 4.16
void retract (
    in eventRetractionHandle     theHandle)
raises (
    InvalidRetractionHandle,
```

```
FederateNotExecutionMember,
RTIinternalError,
UnimplementedService);

///////////////////////////////
// Ownership Management Services //
///////////////////////////////

// 5.1
void requestAttributeOwnershipDivestiture (
    in ObjectId                      theObject,
    in AttributeHandleSet             attrs,
    in OwnershipDivestitureCondition theCondition,
    in UserSuppliedTag               theTag)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidDivestitureCondition,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void requestAttributeOwnershipDivestiture_OVER0 (
    in ObjectId                      theObject,
    in AttributeHandleSet             attrs,
    in OwnershipDivestitureCondition theCondition,
    in UserSuppliedTag               theTag,
    in FederateHandleSet             candidates)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidDivestitureCondition,
    FederateDoesNotExist,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 5.5
void requestAttributeOwnershipAcquisition (
    in ObjectId                      theObject,
    in AttributeHandleSet             attrs,
    in UserSuppliedTag               theTag)
raises (
    ObjectNotKnown,
    ObjectClassNotPublished,
    ObjectClassNotSubscribed,
    AttributeNotDefined,
    AttributeNotPublished,
    AttributeNotSubscribed,
    FederateOwnsAttributes,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);
```

```
// 5.7
void queryAttributeOwnership (
    in ObjectId          theObject,
    in AttributeHandle   theAttr)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 5.9
void isAttributeOwnedByFederate (
    in ObjectId          theObject,
    in AttributeHandle   theAttr,
    out RTI_Boolean      ownershipIndicator)
raises (
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

///////////////////////////////
// Time Management Services //
///////////////////////////////

// 6.1
void requestFederationTime (
    out FederationTime   theTime)
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.2
void requestLBTS (
    out FederationTime   theTime)
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.3
void requestFederateTime (
    out FederationTime   theTime)
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.4
void requestMinNextEventTime (
    out FederationTime   theTime)
```

```
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.5
void setLookahead (
    in FederationTimeDelta      lookahead)
raises (
    InvalidLookaheadTime,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.6
void requestLookahead (
    out FederationTimeDelta     theLookahead)
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.7
void timeAdvanceRequest (
    in FederationTime      theTime)
raises (
    InvalidFederationTime,
    TimeAdvanceAlreadyInProgress,
    FederationTimeAlreadyPassed,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.8
void nextEventRequest (
    in FederationTime      theTime)
raises (
    InvalidFederationTime,
    TimeAdvanceAlreadyInProgress,
    FederationTimeAlreadyPassed,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 6.9

void flushQueueRequest (
    in FederationTime      EtheTime)
raises (
    InvalidFederationTime,
    TimeAdvanceAlreadyInProgress,
    FederationTimeAlreadyPassed,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);
```

```
///////////////////////////////
// Data Distribution Management Services //
///////////////////////////////



// 7.1
void CreateUpdateRegion (
    in SpaceHandle      theSpace,
    in ExtentSet        theExtents,
    out Region          theRegion)
raises (
    SpaceNotDefined,
    InvalidExtents,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 7.2
void CreateSubscriptionRegion (
    in SpaceHandle      theSpace,
    in ExtentSet        theExtents,
    out Region          theRegion)
raises (
    SpaceNotDefined,
    InvalidExtents,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 7.3
void associateUpdateRegion (
    in Region           theRegion,
    in ObjectId          theObject,
    in AttributeHandleSet theAttributes)
raises (
    RegionNotKnown,
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void associateUpdateRegion_OVER0 (
    in Region           theRegion,
    in InteractionClassHandle theClass)
raises (
    RegionNotKnown,
    InteractionClassNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void disassociateUpdateRegion (
    in Region           theRegion,
    in ObjectId          theObject)
```

```
raises (
    RegionNotKnown,
    ObjectNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void disassociateUpdateRegion_OVER0 (
    in Region                  theRegion,
    in InteractionClassHandle   theClass)
raises (
    RegionNotKnown,
    InteractionClassNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 7.5
void modifyRegion (
    in Region      theRegion,
    in ExtentSet   theExtents)
raises (
    RegionNotKnown,
    InvalidExtents,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// 7.6
void deleteRegion (
    in Region      theRegion)
raises (
    RegionNotKnown,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

///////////////////////////////
// the following services are not in the I/F spec
// they are here for informational purposes only
/////////////////////////////
// RTI Support & Operational Services //
/////////////////////////////

// RSOS.1
ObjectClassHandle getObjectClassHandle (
    in ObjectClassName   theClass)
raises (
    NameNotFound,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);
```

```
// RSOS.2
ObjectClassName getObjectClassName (
    in ObjectClassHandle theClass)
raises (
    ObjectClassNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.3
AttributeHandle getAttributeHandle (
    in AttributeName      theAttribute,
    in ObjectClassHandle theClass)
raises (
    ObjectClassNotDefined,
    NameNotFound,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.4
AttributeName getAttributeName (
    in AttributeHandle   theAttribute,
    in ObjectClassHandle theClass)
raises (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.5
InteractionClassHandle getInteractionClassHandle (
    in InteractionClassName theInteraction)
raises (
    NameNotFound,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.6
InteractionClassName getInteractionClassName (
    in InteractionClassHandle theInteraction)
raises (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.7
ParameterHandle getParameterHandle (
    in ParameterName        theParameter,
    in InteractionClassHandle theClass)
raises (
```

```
InteractionClassNotDefined,
NameNotFound,
FederateNotExecutionMember,
RTIinternalError,
UnimplementedService);

// RSOS.8
ParameterName getParameterName (
    in ParameterHandle          theParameter,
    in InteractionClassHandle   theClass)
raises (
    InteractionClassNotDefined,
    InteractionParameterNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.9
SpaceHandle getSpaceHandle (
    in SpaceName   theSpace)
raises (
    NameNotFound,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.10
SpaceName getSpaceName (
    in SpaceHandle      theSpace)
raises (
    SpaceNotDefined,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.11
void turnRegulationOn ()
raises (
    FederationTimeAlreadyPassed,
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

FederationTime turnRegulationOnNow ()
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

void turnRegulationOff ()
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);
```

```
// RSOS.12
void setTimeConstrained (
    in RTI_Boolean      state)
raises (
    FederateNotExecutionMember,
    RTIinternalError,
    UnimplementedService);

// RSOS.13
RTI_boolean tick ()
raises (
    RTIinternalError,
    UnimplementedService);

RTI_boolean tick_OVER0 (
    in TickTime    minimum,
    in TickTime    maximum)
raises (
    RTIinternalError,
    UnimplementedService);

};

#endif
```

```
//File: federateAmbServices.idl

#ifndef __federateAmbServices
#define __federateAmbServices

#include "rtiTYPES.idl"

///////////////////////////////
// Federation Management Services //
///////////////////////////////

interface federateAmbassador: rtiTypes {

    // 2.6
    void initiatePause (
        in PauseLabel label)
    raises (
        FederateAlreadyPaused,
        FederateInternalError);

    // 2.9
    void initiateResume ( )
    raises (
        FederateNotPaused,
        FederateInternalError);

    // 2.12
    void initiateFederateSave (
        in SaveLabel label)
    raises (
        UnableToPerformSave,
        FederateInternalError);

    void initiateFederateSave_OVER0 (
        in SaveLabel          label,
        in FederationTime    theTime)
    raises (
        UnableToPerformSave,
        InvalidFederationTime,
        FederateInternalError);

    // 2.16
    void initiateRestore (
        in SaveLabel label)
    raises (
        SpecifiedSaveLabelDoesNotExist,
        CouldNotInitiateRestore,
        FederateInternalError);

/////////////////////////////
// Declaration Management Services //
/////////////////////////////

    // 3.5
```

```
void startUpdates (
    in ObjectClassHandle theClass,
    in AttributeHandleSet attributeList)
raises (
    ObjectClassNotPublished,
    AttributeNotPublished,
    FederateInternalError);

void stopUpdates (
    in ObjectClassHandle theClass,
    in AttributeHandleSet attributeList)
raises (
    ObjectClassNotPublished,
    AttributeNotPublished,
    FederateInternalError);

// 3.6
void startInteractionGeneration (
    in InteractionClassHandle theClass)
raises (
    InteractionClassNotPublished,
    FederateInternalError);

void stopInteractionGeneration (
    in InteractionClassHandle theClass)
raises (
    InteractionClassNotPublished,
    FederateInternalError);

///////////////////////////////
// Object Management Services //
///////////////////////////////

// 4.4
void discoverObject (
    in ObjectId                  theObject,
    in ObjectClassHandle          theClass,
    in FederationTime            theTime,
    in UserSuppliedTag           theTag,
    in EventRetractionHandle     theHandle)
raises (
    CouldNotDiscover,
    ObjectClassNotKnown,
    InvalidFederationTime,
    FederateInternalError);

// 4.5
void reflectAttributeValuePairs (
    in ObjectId                  objectId,
    in AttributeHandleValuePairSet attributeValueList,
    in FederationTime            theTime,
    in UserSuppliedTag           theTag,
    in EventRetractionHandle     theHandle)
raises (
    ObjectNotKnown,
```

```
AttributeNotKnown,
InvalidFederationTime,
FederateInternalError);

// 4.7
void receiveInteraction (
    in InteractionClassHandle          theInteraction,
    in ParameterHandleValuePairSet     parameters,
    in FederationTime                 theTime,
    in UserSuppliedTag                theTag,
    in EventRetractionHandle          theHandle)
raises (
    InteractionClassNotFound,
    InteractionParameterNotFound,
    InvalidFederationTime,
    FederateInternalError);

// 4.9
void removeObject (
    in ObjectId                      theObject,
    in ObjectRemovalReason           theReason,
    in FederationTime                theTime,
    in UserSuppliedTag               theTag,
    in EventRetractionHandle          theHandle)
raises (
    ObjectNotFound,
    InvalidFederationTime,
    FederateInternalError);

void removeObject_OVER (
    in ObjectId                      theObject,
    in ObjectRemovalReason           theReason,
    in FederationTime                theTime,
    in UserSuppliedTag               theTag)
raises (
    ObjectNotFound,
    InvalidFederationTime,
    FederateInternalError);

// 4.15
void provideAttributeValueUpdate (
    in ObjectId                      theObject,
    in AttributeHandleSet            attributeList)
raises (
    ObjectNotFound,
    AttributeNotFound,
    FederateInternalError);

// 4.17
void reflectRetraction (
    in EventRetractionHandle         theHandle)
raises (
    EventNotFound,
    FederateInternalError);
```

```
///////////////////////////////
// Ownership Management Services //
///////////////////////////////

// 5.2
void requestAttributeOwnershipAssumption (
    in ObjectId                  theObject,
    in AttributeHandleSet        offeredAttributes,
    in UserSuppliedTag          theTag,
    out AttributeHandleSet       requestedAttributes)
raises (
    ObjectNotKnown,
    AttributeNotKnown,
    AttributeAlreadyOwned,
    FederateInternalError);

// 5.3
void attributeOwnershipDivestitureNotification (
    in ObjectId                  theObject,
    in AttributeHandleSet        releasedAttributes)
raises (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError);

// 5.4
void attributeOwnershipAcquisitionNotification (
    in ObjectId                  theObject,
    in AttributeHandleSet        securedAttributes)
raises (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError);

// 5.6
void requestAttributeOwnershipRelease (
    in ObjectId                  theObject,
    in AttributeHandleSet        candidateAttributes,
    in UserSuppliedTag          theTag,
    out AttributeHandleSet       releasedAttributes)
raises (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError);

// 5.8
void informAttributeOwnership (
    in ObjectId                  theObject,
    in AttributeHandle           theAttribute,
    in FederateHandle            theOwner)
raises (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError);
```

```
/////////////////////////////
// Time Management Services //
/////////////////////////////

// 6.10
void timeAdvanceGrant (
    in FederationTime theTime)
raises (
    InvalidFederationTime,
    TimeAdvanceNotInProgress,
    FederationTimeAlreadyPassed,
    FederateInternalError);

/////////////////////////////
// Data Distribution Management Services //
/////////////////////////////

// 7.4
void changeThresholds (
    in Region          theRegion,
    in ThresholdSet    theThresholds)
raises (
    RegionNotKnown,
    FederateInternalError);

};

#endif
```

9 HLA C++ Application Programmer's Interface

```
//File baseTypes.hh

#ifndef NULL
#define NULL (0)
#endif

typedef unsigned short UShort;
typedef short Short;
typedef unsigned long ULONG;
typedef long Long;
typedef double Double;
typedef float Float;

enum Boolean {
    RTI_FALSE = 0,
    RTI_TRUE};

class Exception {
public:
    RTI::ULONG _serial;
    char *_reason;
    const char *_name;
    Exception (const char *reason);
    Exception (RTI::ULONG serial, const char *reason=NULL);
    Exception (const Exception &toCopy);
    virtual ~Exception ();
    Exception & operator = (const Exception &);

    friend ostream& operator<< (ostream &, Exception *);
};

#define RTI_EXCEPT(A) \
class A : public RTI::Exception {           \
public: \
    static const char *_ex; \
    A (const char *reason) : Exception (reason) { _name = _ex; \
}   \
    A (RTI::ULONG serial, const char *reason=NULL)           \
        : Exception (serial, reason) { _name = _ex; }           \
    A (const RTI::Exception &toCopy) : Exception(toCopy) { \
_name = _ex; } \
};


```

```
//File: RTItypes.hh

//          RTI Parameter Passing Memory Conventions
//

// C1  In parameter by value.
// C2  Out parameter by reference.
// C3  Function return by value.
// C4  In parameter by const reference.  Caller provides memory.
//      Caller may free memory or overwrite it upon completion of
//      the call.  Callee must copy during the call anything it
//      wishes to save beyond completion of the call.  Parameter
//      type must define const accessor methods.
// C5  Out parameter by reference.  Caller provides reference to object.
//      Callee constructs an instance on the heap (new) and returns.
//      The caller destroys the instance (delete) at its leisure.
// C6  Function return by reference.  Callee constructs an instance on
//      the heap (new) and returns a reference.  The caller destroys the
//      instance (delete) at its leisure.

#define MAX_FEDERATION          20
#define MAX_FEDERATE             20
#define MAX_FEDERATION_NAME_LENGTH 64
#define MAX_FEDERATE_NAME_LENGTH 64
#define MAX_OBJECT_CLASSES        100
#define MAX_INTERACTION_CLASSES   100
#define MAX_ATTRIBUTES_PER_CLASS 50
#define MAX_PARAMETERS_PER_CLASS 50
#define MAX_BYTES_PER_VALUE       512
#define PRIVILEGE_TO_DELETE_HANDLE 0
#define ROOT_OBJECT_CLASS_HANDLE 0
#define ROOT_INTERACTION_CLASS_HANDLE 0
#define ROOT_OBJECT_CLASS_NAME    "root"
#define ROOT_INTERACTION_CLASS_NAME "root"
#define PRIVILEGE_TO_DELETE_NAME  "privilegeToDelete"
#define MAX_USER_TAG_LENGTH       16
#define RTI_VERSION                "1.0R2"

#define EPSILON (1.0e-9)
//as defined for IEEE 754
//see <math.h>
#define POSITIVE_INFINITY (HUGE_VAL)

RTI_EXCEPT(ArrayIndexOutOfBoundsException)
RTI_EXCEPT(AttributeAlreadyOwned)
RTI_EXCEPT(AttributeNotDefined)
RTI_EXCEPT(AttributeNotKnown)
RTI_EXCEPT(AttributeNotOwned)
RTI_EXCEPT(AttributeNotPublished)
RTI_EXCEPT(AttributeNotSubscribed)
RTI_EXCEPT(ConcurrentAccessAttempted)
RTI_EXCEPT(CouldNotDiscover)
RTI_EXCEPT(CouldNotOpenFED)
RTI_EXCEPT(CouldNotRestore)
RTI_EXCEPT(DeletePrivilegeNotHeld)
```

```
RTI_EXCEPT(ErrorReadingFED)
RTI_EXCEPT(EventNotKnown)
RTI_EXCEPT(FederateAlreadyPaused)
RTI_EXCEPT(FederateAlreadyExecutionMember)
RTI_EXCEPT(FederateDoesNotExist)
RTI_EXCEPT(FederateInternalError)
RTI_EXCEPT(FederateNameAlreadyInUse)
RTI_EXCEPT(FederateNotExecutionMember)
RTI_EXCEPT(FederateNotPaused)
RTI_EXCEPT(FederateOwnsAttributes)
RTI_EXCEPT(FederatesCurrentlyJoined)
RTI_EXCEPT(FederationAlreadyPaused)
RTI_EXCEPT(FederationExecutionAlreadyExists)
RTI_EXCEPT(FederationExecutionDoesNotExist)
RTI_EXCEPT(FederationNotPaused)
RTI_EXCEPT(FederationTimeAlreadyPassed)
RTI_EXCEPT(HandleValuePairMaximumExceeded)
RTI_EXCEPT(IDsupplyExhausted)
RTI_EXCEPT(InteractionClassNotDefined)
RTI_EXCEPT(InteractionClassNotKnown)
RTI_EXCEPT(InteractionClassNotPublished)
RTI_EXCEPT(InteractionParameterNotDefined)
RTI_EXCEPT(InteractionParameterNotKnown)
RTI_EXCEPT(InvalidDivestitureCondition)
RTI_EXCEPT(InvalidExtents)
RTI_EXCEPT(InvalidFederationTime)
RTI_EXCEPT(InvalidLookahead)
RTI_EXCEPT(InvalidObjectID)
RTI_EXCEPT(InvalidOrderType)
RTI_EXCEPT(InvalidResignAction)
RTI_EXCEPT(InvalidRetractionHandle)
RTI_EXCEPT(InvalidTransportType)
RTI_EXCEPT(MemoryExhausted)
RTI_EXCEPT(NameNotFound)
RTI_EXCEPT(NoPauseRequested)
RTI_EXCEPT(NoResumeRequested)
RTI_EXCEPT(ObjectClassNotDefined)
RTI_EXCEPT(ObjectClassNotKnown)
RTI_EXCEPT(ObjectClassNotPublished)
RTI_EXCEPT(ObjectClassNotSubscribed)
RTI_EXCEPT(ObjectNotKnown)
RTI_EXCEPT(ObjectAlreadyRegistered)
RTI_EXCEPT(RegionNotKnown)
RTI_EXCEPT(RestoreNotRequested)
RTI_EXCEPT(RTIinternalError)
RTI_EXCEPT(SpaceNotDefined)
RTI_EXCEPT(SaveInProgress)
RTI_EXCEPT(SaveNotInitiated)
RTI_EXCEPT(SpecifiedSaveLabelDoesNotExist)
RTI_EXCEPT(TimeAdvanceAlreadyInProgress)
RTI_EXCEPT(TimeAdvanceWasNotInProgress)
RTI_EXCEPT(TooManyIDsRequested)
RTI_EXCEPT(UnableToPerformSave)
RTI_EXCEPT(UnimplementedService)
RTI_EXCEPT(UnknownLabel)
```

```
RTI_EXCEPT(ValueCountExceeded)
RTI_EXCEPT(ValueLengthExceeded)

enum OrderType {
    RECEIVE = 1,
    TIMESTAMP
};

enum OwnershipDivestitureCondition {
    NEGOTIATED = 1,
    UNCONDITIONAL
};

enum ObjectRemovalReason {
    OUT_OF_REGION = 1,
    OBJECT_DELETED,
    NO_LONGER_SUBSCRIBED
};

enum TransportType {
    RELIABLE = 1,
    BEST EFFORT
};

enum ResignAction {
    RELEASE_ATTRIBUTES = 1,
    DELETE_OBJECTS,
    DELETE_OBJECTS_AND_RELEASE_ATTRIBUTES,
    NO_ACTION
};

class FederateAmbassador;

typedef FederateAmbassador *FederateAmbassadorPtr;

typedef RTI::UShort ObjectClassHandle;

typedef RTI::UShort InteractionClassHandle;

typedef RTI::UShort Handle;

typedef Handle AttributeHandle;

typedef Handle ParameterHandle;

typedef RTI::UShort FederateHandle;

typedef RTI::ULong FederateID;

typedef RTI::ULong ObjectID;

typedef RTI::ULong UniqueID;

class HandleValuePairSet {
public:
```

```
virtual ~HandleValuePairSet() { ; }

virtual ULONG      // returned C3
size() const = 0; // Cardinality of the set.

virtual Handle // returned C3
getHandle(RTI::ULONG i) const
throw (
    ArrayIndexOutOfBoundsException) = 0;

virtual void getValue(RTI::ULONG i,
    char *buff,           // supplied C4
    ULONG &valueLength) const // returned C2
throw (
    ArrayIndexOutOfBoundsException) = 0;

virtual void add(      // Add new value.
    Handle h,           // supplied C1
    const char *buff,   // supplied C4
    ULONG valueLength) // supplied C1
throw (
    ValueLengthExceeded,
    ValueCountExceeded) = 0;

virtual void moveFrom(const HandleValuePairSet& ahvp,
    RTI::ULONG &i) // supplied C1
throw (
    ValueCountExceeded,
    ArrayIndexOutOfBoundsException) = 0;

virtual void empty() = 0; // Empty the Set without deallocating space.

virtual inline RTI::ULONG start() const = 0;
virtual inline RTI::ULONG valid(RTI::ULONG i) const = 0;
virtual inline RTI::ULONG next(RTI::ULONG i) const = 0;
};

typedef HandleValuePairSet AttributeHandleValuePairSet;

class AttributeSetFactory {
public:
    static AttributeHandleValuePairSet // returned C6
    *create(                         // Create set with space for
                                    // specified number of attributes.
        ULONG count)                // supplied C1
    throw (
        MemoryExhausted,
        ValueCountExceeded);
};

class AttributeHandleSet {
public:
    AttributeHandleSet();
    ULONG      // returned C3
size() const; // Cardinality of the set.
```

```
AttributeHandle // returned C3
getHandle(          // Get ith handle.
    ULong i) const // supplied C1
throw (
    ArrayIndexOutOfBoundsException);

void add( // Add new value.
    AttributeHandle h) // supplied C1
throw (
    AttributeNotDefined);

void empty(); // Empty the Set

Boolean isEmpty(); //is set empty?
Boolean isMember(AttributeHandle h) const;
AttributeHandleSet setUnion(AttributeHandleSet &);
AttributeHandleSet setIntersection(AttributeHandleSet &);
AttributeHandleSet removeSetIntersection(AttributeHandleSet &);

ULong encodedLength() const;
void encode(char *buff) const;
static AttributeHandleSet *decode(const char *buff)
    throw(ValueCountExceeded);

friend ostream& operator << (ostream&, AttributeHandleSet &);

private:
unsigned short _size;
unsigned long _words[(MAX_ATTRIBUTES_PER_CLASS-1)/32+1];
};

class AttributeHandleSetFactory {
public:
    static AttributeHandleSet // returned C6
    *create(                  // Create set with space for specified
                            // number of AttributeHandles.
        ULong count)         // supplied C1
    throw(
        MemoryExhausted,
        ValueCountExceeded);
};

typedef RTI::UShort ObjectIDcount;

typedef RTI::Double FederationTimeDelta;

typedef RTI::Double FederationTime;

typedef RTI::Double TickTime;

class FederateHandleSet {
public:
```

```
virtual ULONG      // returned C3
size() const = 0; // Cardinality of the set.

virtual FederateHandle // returned C3
getHandle(           // Get ith handle.
    ULONG i) const // supplied C1
throw (
    ArrayIndexOutOfBoundsException) = 0;

virtual void add(   // Add new value.
    FederateHandle h) // supplied C1
throw (
    ValueCountExceeded) = 0;

virtual void empty() = 0; // Empty the set without deallocating space.
};

class FederateHandleSetFactory {
public:
    static FederateHandleSet // returned C6
    *create(                // Create set with space for specified
                           // number of FederateHandles.
        ULONG count)       // supplied C1
    throw (
        MemoryExhausted,
        ValueCountExceeded);
};

struct EventRetractionHandle_s {
    FederationTime theEventTime;
    UniqueID theSerialNumber;
    FederateHandle sendingFederate;
};
typedef struct EventRetractionHandle_s EventRetractionHandle;

// All char * declarations should use null terminated strings

typedef char * FederationExecutionName;

typedef char * FederateName;

typedef char * FileName;

typedef char * PauseLabel;

typedef char * SaveLabel;

typedef char * UserSuppliedTag;

typedef char * ObjectClassName;

typedef char * AttributeName;

typedef char * InteractionClassName;
```

```
typedef char * ParameterName;

typedef HandleValuePairSet ParameterHandleValuePairSet;

class ParameterSetFactory {
public:
    static ParameterHandleValuePairSet // returned C6
    *create(                                // Create set with space for
                                              // specified number of attributes.
        ULong count)                      // supplied C1
    throw (
        MemoryExhausted,
        ValueCountExceeded);
};

struct range_struct_s {
    RTI::ULong first;
    RTI::ULong last;
};
typedef struct range_struct_s Range;

struct Extents_s {
    RTI::ULong _length;
    Range* _buffer;
};
typedef struct Extents_s Extents;

struct ExtentSet_s {
    RTI::ULong _length;
    Extents* _buffer;
};
typedef struct ExtentSet_s ExtentSet;

typedef RTI::ULong Region;

struct Threshold_s {
    RTI::ULong value;
};
typedef Threshold_s Threshold;

struct ThresholdSet_s {
    RTI::ULong _length;
    Threshold* _buffer;
};
typedef struct ThresholdSet_s ThresholdSet;

typedef RTI::Long SpaceHandle;

typedef char * SpaceName;
```

```
//File: RTIambServices.hh

//          RTI Parameter Passing Memory Conventions
//
// C1  In parameter by value.
// C2  Out parameter by reference.
// C3  Function return by value.
// C4  In parameter by const reference.  Caller provides memory.
//     Caller may free memory or overwrite it upon completion of
//     the call.  Callee must copy during the call anything it
//     wishes to save beyond completion of the call.  Parameter
//     type must define const accessor methods.
// C5  Out parameter by reference.  Caller provides reference to object.
//     Callee constructs an instance on the heap (new) and returns.
//     The caller destroys the instance (delete) at its leisure.
// C6  Function return by reference.  Callee constructs an instance on
//     the heap (new) and returns a reference.  The caller destroys the
//     instance (delete) at its leisure.

typedef FederateAmbassador *FederateAmbassadorPtr;

///////////////////////////////
// Federation Management Services //
///////////////////////////////

// 2.1
void createFederationExecution (
    const FederationExecutionName executionName) // supplied C4
throw (
    FederationExecutionAlreadyExists,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 2.2
void destroyFederationExecution (
    const FederationExecutionName executionName) // supplied C4
throw (
    FederatesCurrentlyJoined,
    FederationExecutionDoesNotExist,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 2.3
FederateHandle
joinFederationExecution (                                // returned C3
    const FederateName           yourName,             // supplied C4
    const FederationExecutionName executionName,        // supplied C4
    FederateAmbassadorPtr      federateAmbassadorReference) // supplied C1
throw (
    FederateAlreadyExecutionMember,
    FederationExecutionDoesNotExist,
    CouldNotOpenFED,
    ErrorReadingFED,
    ConcurrentAccessAttempted,
```

```
RTIinternalError);

// 2.4
void resignFederationExecution (
    ResignAction theAction) // supplied C1
throw (
    FederateOwnsAttributes,
    FederateNotExecutionMember,
    FederationExecutionDoesNotExist,
    InvalidResignAction,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 2.5
void requestPause (      // not implemented in F.0
    const PauseLabel label) // supplied C4
throw (
    FederationAlreadyPaused,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.7
void pauseAchieved (      // not implemented in F.0
    const PauseLabel label) // supplied C4
throw (
    UnknownLabel,
    NoPauseRequested,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.8
void requestResume () // not implemented in F.0
throw (
    FederationNotPaused,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.10
void resumeAchieved () // not implemented in F.0
throw (
    NoResumeRequested,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.11
void requestFederationSave ( // not implemented in F.0
    const SaveLabel      label, // supplied C4
```

```
    FederationTime theTime) // supplied C1
throw (
    FederationTimeAlreadyPassed,
    InvalidFederationTime,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void requestFederationSave ( // not implemented in F.0
    const SaveLabel label)      // supplied C4
throw (
    SaveInProgress,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.13
void federateSaveBegun () // not implemented in F.0
throw (
    SaveNotInitiated,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void federateSaveBegun ( // not implemented in F.0
    FederationTime theTime) // supplied C1
throw (
    SaveNotInitiated,
    InvalidFederationTime,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.14
void federateSaveAchieved () // not implemented in F.0
throw (
    SaveNotInitiated,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void federateSaveNotAchieved () // not implemented in F.0
throw (
    SaveNotInitiated,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.15
```

```
void requestRestore (    // not implemented in F.0
    const SaveLabel label) // supplied C4
throw (
    SpecifiedSaveLabelDoesNotExist,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 2.17
void restoreAchieved (    // not implemented in F.0
    const SaveLabel label) // supplied C4
throw (
    UnknownLabel,
    RestoreNotRequested,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void restoreNotAchieved ( // not implemented in F.0
    const SaveLabel label) // supplied C4
throw (
    UnknownLabel,
    RestoreNotRequested,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

///////////////////////////////
// Declaration Management Services //
///////////////////////////////

// 3.1
void publishObjectClass (
    ObjectClassHandle theClass,      // supplied C1
    const AttributeHandleSet& attributeList) // supplied C4
throw (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void unpublishObjectClass (
    ObjectClassHandle theClass) // supplied C1
throw (
    ObjectClassNotDefined,
    FederateOwnsAttributes,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 3.2
```

```
void publishInteractionClass (
    InteractionClassHandle theInteraction) // supplied C1
throw (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void unpublishInteractionClass (
    InteractionClassHandle theInteraction) // supplied C1
throw (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 3.3
void subscribeObjectClassAttribute (
    ObjectClassHandle theClass,          // supplied C1
    const AttributeHandleSet& attributeList) // supplied C4
throw (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void subscribeObjectClassAttribute ( // not implemented in F.0
    ObjectClassHandle theClass,          // supplied
    AttributeHandle theAttribute,        // supplied
    Region theRegion)                  // supplied
throw (
    ObjectClassNotDefined,
    AttributeNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void unsubscribeObjectClassAttribute (
    ObjectClassHandle theClass) // supplied C1
throw (
    ObjectClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void unsubscribeObjectClassAttribute ( // not implemented in F.0
    ObjectClassHandle theClass,          // supplied
    Region theRegion)                  // supplied
throw (
    ObjectClassNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
```

```
ConcurrentAccessAttempted,
RTIinternalError,
UnimplementedService);

// 3.4
void subscribeInteractionClass (
    InteractionClassHandle theClass) // supplied C1
throw (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void subscribeInteractionClass (      // not implemented in F.0
    InteractionClassHandle theClass, // supplied
    Region                  theRegion) // supplied
throw (
    InteractionClassNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void unsubscribeInteractionClass (
    InteractionClassHandle theClass) // supplied C1
throw (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void unsubscribeInteractionClass ( // not implemented in F.0
    InteractionClassHandle theClass, // supplied
    Region                  theRegion) // supplied
throw (
    InteractionClassNotDefined,
    RegionNotKnown,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

///////////////////////////////
// Object Management Services //
///////////////////////////////

// 4.1
void requestID (
    ObjectIDcount idCount, // supplied C1
    ObjectID&     firstID, // returned C2
    ObjectID&     lastID) // returned C2
throw (
    TooManyIDsRequested,
    IDsupplyExhausted,
```

```
FederateNotExecutionMember,
ConcurrentAccessAttempted,
RTIinternalError);

// 4.2
void registerObject (
    ObjectClassHandle theClass, // supplied C1
    ObjectID          theObject) // supplied C1
throw (
    InvalidObjectID,
    ObjectAlreadyRegistered,
    ObjectClassNotDefined,
    ObjectClassNotPublished,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.3
EventRetractionHandle           // returned C3
updateAttributeValues (
    ObjectID          theObject, // supplied C1
    const AttributeHandleValuePairSet& theAttributes, // supplied C4
    FederationTime      theTime, // supplied C1
    const UserSuppliedTag   theTag) // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidFederationTime,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.6
EventRetractionHandle           // returned C3
sendInteraction (
    InteractionClassHandle     theInteraction, // supplied C1
    const ParameterHandleValuePairSet& theParameters, // supplied C4
    FederationTime      theTime, // supplied C1
    const UserSuppliedTag   theTag) // supplied C4
throw (
    InteractionClassNotPublished,
    InteractionClassNotDefined,
    InteractionParameterNotDefined,
    InvalidFederationTime,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.8
EventRetractionHandle           // returned C3
deleteObject (
    ObjectID          objectID, // supplied C1
    FederationTime      theTime, // supplied C1
    const UserSuppliedTag   theTag) // supplied C4
```

```
throw (
    DeletePrivilegeNotHeld,
    ObjectNotKnown,
    InvalidFederationTime,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.10
void changeAttributeTransportType (
    ObjectId          theObject,      // supplied C1
    const AttributeHandleSet& theAttributes, // supplied C4
    TransportType       theType)        // supplied C1
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidTransportType,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.11
void changeAttributeOrderType (
    ObjectId          theObject,      // supplied C1
    const AttributeHandleSet& theAttributes, // supplied C4
    OrderType         theType)        // supplied C1
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidOrderType,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.12
void changeInteractionTransportType (
    InteractionClassHandle theClass, // supplied C1
    TransportType           theType) // supplied C1
throw (
    InteractionClassNotDefined,
    InteractionClassNotPublished,
    InvalidTransportType,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.13
void changeInteractionOrderType (
    InteractionClassHandle theClass, // supplied C1
    OrderType              theType) // supplied C1
throw (
    InteractionClassNotDefined,
    InteractionClassNotPublished,
```

```
    InvalidOrderType,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.14
void requestObjectAttributeValueUpdate (
    ObjectId             theObject,      // supplied C1
    const AttributeHandleSet& theAttributes) // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void requestObjectAttributeValueUpdate (
    ObjectId theObject) // supplied C1
throw (
    ObjectNotKnown,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void requestClassAttributeValueUpdate (
    ObjectClassHandle   theClass,        // supplied C1
    const AttributeHandleSet& theAttributes) // supplied C4
throw (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void requestClassAttributeValueUpdate (
    ObjectClassHandle theClass) // supplied C1
throw (
    ObjectClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 4.16
void retract (
    EventRetractionHandle theHandle) // supplied C1
throw (
    InvalidRetractionHandle,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

///////////////////////////////
// Ownership Management Services //
///////////////////////////////
```

```
// 5.1
void requestAttributeOwnershipDivestiture (
    ObjectId             theObject,      // supplied C1
    const AttributeHandleSet& theAttributes, // supplied C4
    OwnershipDivestitureCondition theCondition, // supplied C1
    const UserSuppliedTag       theTag)        // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidDivestitureCondition,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

void requestAttributeOwnershipDivestiture (
    ObjectId             theObject,      // supplied C1
    const AttributeHandleSet& theAttributes, // supplied C4
    OwnershipDivestitureCondition theCondition, // supplied C1
    const UserSuppliedTag       theTag,        // supplied C4
    const FederateHandleSet&   theCandidates) // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotOwned,
    InvalidDivestitureCondition,
    FederateDoesNotExist,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 5.5
void requestAttributeOwnershipAcquisition (
    ObjectId             theObject,      // supplied C1
    const AttributeHandleSet& desiredAttributes, // supplied C4
    const UserSuppliedTag       theTag)        // supplied C4
throw (
    ObjectNotKnown,
    ObjectClassNotPublished,
    ObjectClassNotSubscribed,
    AttributeNotDefined,
    AttributeNotPublished,
    AttributeNotSubscribed,
    FederateOwnsAttributes,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 5.7
void queryAttributeOwnership (
    ObjectId             theObject,      // supplied C1
    AttributeHandle theAttribute) // supplied C1
throw (
    ObjectNotKnown,
    AttributeNotDefined,
```

```
FederateNotExecutionMember,
ConcurrentAccessAttempted,
RTIinternalError);

// 5.9
RTI::Boolean // returned C3
attributeIsOwnedByFederate (
    ObjectID      theObject,      // supplied C1
    AttributeHandle theAttribute) // supplied C1
throw (
    ObjectNotKnown,
    AttributeNotDefined,
    AttributeNotKnown,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

///////////////////////////////
// Time Management Services //
///////////////////////////////

// 6.1
FederationTime // returned C3
requestFederationTime ()
throw (
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.2
FederationTime // returned C3
requestLBTS ()
throw (
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.3
FederationTime // returned C3
requestFederateTime ()
throw (
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.4
FederationTime // returned C3
requestMinNextEventTime ()
throw (
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.5
void setLookahead (
```

```
FederationTimeDelta theLookahead) // supplied C1
throw (
    InvalidLookahead,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.6
FederationTimeDelta // returned C3
requestLookahead ()
throw (
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.7
void timeAdvanceRequest (
    FederationTime theTime) // supplied C1
throw (
    InvalidFederationTime,
    TimeAdvanceAlreadyInProgress,
    FederationTimeAlreadyPassed,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.8
void nextEventRequest (
    FederationTime theTime) // supplied C1
throw (
    TimeAdvanceAlreadyInProgress,
    FederationTimeAlreadyPassed,
    InvalidFederationTime,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 6.9
void flushQueueRequest (
    FederationTime theTime) // supplied C1
throw (
    InvalidFederationTime,
    TimeAdvanceAlreadyInProgress,
    FederationTimeAlreadyPassed,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

///////////////////////////////
// Data Distribution Management //
///////////////////////////////

// 7.1
void createUpdateRegion ( // not implemented in F.0
    SpaceHandle theSpace, // supplied
```

```
ExtentSet& theExtents, // supplied
Region& theRegion) // returned
throw (
    SpaceNotDefined,
    InvalidExtents,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 7.2
void createSubscriptionRegion ( // not implemented in F.0
    SpaceName theSpace,           // supplied
    ExtentSet& theExtents,        // supplied
    Region& theRegion)          // returned
throw (
    SpaceNotDefined,
    InvalidExtents,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 7.3
void associateUpdateRegion (           // not implemented in F.0
    Region          theRegion,      // supplied
    ObjectID         theObject,      // supplied
    AttributeHandleSet& theAttributes) // supplied
throw (
    RegionNotKnown,
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void disassociateUpdateRegion (         // not implemented in F.0
    Region          theRegion,      // supplied
    ObjectID         theObject,      // supplied
    AttributeHandleSet& theAttributes) // supplied
throw (
    RegionNotKnown,
    ObjectNotKnown,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

void associateUpdateRegion (           // not implemented in F.0
    Region          theRegion,      // supplied
    InteractionClassHandle theClass) // supplied
throw (
    RegionNotKnown,
```

```
InteractionClassNotKnown,
FederateNotExecutionMember,
ConcurrentAccessAttempted,
RTIinternalError,
UnimplementedService);

void disassociateUpdateRegion (      // not implemented in F.0
    Region           theRegion, // supplied
    InteractionClassHandle theClass) // supplied
throw (
    RegionNotFound,
    InteractionClassNotKnown,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 7.5
void modifyRegion (      // not implemented in F.0
    Region       theRegion, // supplied
    ExtentSet& theExtents) // supplied
throw (
    RegionNotFound,
    InvalidExtents,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 7.6
void deleteRegion ( // not implemented in F.0
    Region theRegion) // supplied
throw (
    RegionNotFound,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

///////////////////////////////
// RTI Support Services //
////////////////////////////

// 8.1
ObjectClassHandle          // returned C3
getObjectClassHandle (
    const ObjectClassName theName) // supplied C4
throw (
    NameNotFound,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.2
ObjectClassName             // returned C6
```

```
getObjectName (
    ObjectClassHandle theHandle) // supplied C1
throw (
    ObjectClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.3
AttributeHandle // returned C3
getAttributeHandle (
    const AttributeName theName, // supplied C4
    ObjectClassHandle whichClass) // supplied C1
throw (
    ObjectClassNotDefined,
    NameNotFound,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.4
AttributeName // returned C6
getAttributeName (
    AttributeHandle theHandle, // supplied C1
    ObjectClassHandle whichClass) // supplied C1
throw (
    ObjectClassNotDefined,
    AttributeNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.5
InteractionClassHandle // returned C1
getInteractionClassHandle (
    const InteractionClassName theName) // supplied C4
throw (
    NameNotFound,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.6
InteractionClassName // returned C6
getInteractionClassName (
    InteractionClassHandle theHandle) // supplied C1
throw (
    InteractionClassNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.7
ParameterHandle // returned C1
getParameterHandle (
```

```
const ParameterName          theName,      // supplied C4
      InteractionClassHandle whichClass) // supplied C1
throw (
    InteractionClassNotDefined,
    NameNotFound,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.8
ParameterName          // returned C6
getParameterName (
    ParameterHandle      theHandle,    // supplied C1
    InteractionClassHandle whichClass) // supplied C1
throw (
    InteractionClassNotDefined,
    InteractionParameterNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.9
SpaceHandle          // returned C1
getSpaceHandle (        // not implemented in F.0
    const SpaceName theName) // supplied C4
throw (
    NameNotFound,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 8.10
SpaceName          // returned C6
getSpaceName (        // not implemented in F.0
    const SpaceHandle theHandle) // supplied C4
throw (
    SpaceNotDefined,
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError,
    UnimplementedService);

// 8.11
void RTI::RTIambassador::turnRegulationOn ()
throw (
    RTI::FederationTimeAlreadyPassed,
    RTI::FederateNotExecutionMember,
    RTI::ConcurrentAccessAttempted,
    RTI::RTIinternalError);

RTI::FederationTime          // Returned C3
RTI::RTIambassador::turnRegulationOnNow ()
throw (
    RTI::FederateNotExecutionMember,
```

```
RTI::ConcurrentAccessAttempted,
RTI::RTIinternalError);

void RTI::RTIambassador::turnRegulationOff ()
throw (
    RTI::FederateNotExecutionMember,
    RTI::ConcurrentAccessAttempted,
    RTI::RTIinternalError);

// 8.12
void setTimeConstrained (
    RTI::Boolean state) // supplied C1
throw (
    FederateNotExecutionMember,
    ConcurrentAccessAttempted,
    RTIinternalError);

// 8.13
RTI::Boolean // returned C3
tick ()
throw (
    ConcurrentAccessAttempted,
    RTIinternalError);

RTI::Boolean      // returned C3
tick (
    TickTime minimum, // supplied C1
    TickTime maximum) // supplied C1
throw (
    ConcurrentAccessAttempted,
    RTIinternalError);

RTIambassador()
throw (
    MemoryExhausted,
    RTIinternalError);

~RTIambassador()
throw (RTIinternalError);
```

```
//File: federateAmbServices.hh

//          RTI Parameter Passing Memory Conventions
//
// C1  In parameter by value.
// C2  Out parameter by reference.
// C3  Function return by value.
// C4  In parameter by const reference.  Caller provides memory.
//      Caller may free memory or overwrite it upon completion of
//      the call.  Callee must copy during the call anything it
//      wishes to save beyond completion of the call.  Parameter
//      type must define const accessor methods.
// C5  Out parameter by reference.  Caller provides reference to object.
//      Callee constructs an instance on the heap (new) and returns.
//      The caller destroys the instance (delete) at its leisure.
// C6  Function return by reference.  Callee constructs an instance on
//      the heap (new) and returns a reference.  The caller destroys the
//      instance (delete) at its leisure.

///////////////////////////////
// Federation Management Services //
///////////////////////////////

// 2.6
virtual void initiatePause (
    const PauseLabel label) // supplied C4
throw (
    FederateAlreadyPaused,
    FederateInternalError) = 0;

// 2.9
virtual void initiateResume ()
throw (
    FederateNotPaused,
    FederateInternalError) = 0;

// 2.12
virtual void initiateFederateSave (
    const SaveLabel label) // supplied C4
throw (
    UnableToPerformSave,
    FederateInternalError) = 0;

virtual void initiateFederateSave (
    const SaveLabel      label,    // supplied C4
    FederationTime theTime) // supplied C1
throw (
    InvalidFederationTime,
    UnableToPerformSave,
    FederateInternalError) = 0;

// 2.16
virtual void initiateRestore (
    const SaveLabel label) // supplied C4
```

```
throw (
    SpecifiedSaveLabelDoesNotExist,
    CouldNotRestore,
    FederateInternalError) = 0;

///////////////////////////////
// Declaration Management Services //
///////////////////////////////

// 3.5
virtual void startUpdates (
    ObjectClassHandle theClass,          // supplied C1
    const AttributeHandleSet& theAttributes) // supplied C4
throw (
    ObjectClassNotPublished,
    AttributeNotPublished,
    FederateInternalError) = 0;

virtual void stopUpdates (
    ObjectClassHandle theClass,          // supplied C1
    const AttributeHandleSet& theAttributes) // supplied C4
throw (
    ObjectClassNotPublished,
    AttributeNotPublished,
    FederateInternalError) = 0;

// 3.6
virtual void startInteractionGeneration (
    InteractionClassHandle theHandle) // supplied C1
throw (
    InteractionClassNotPublished,
    FederateInternalError) = 0;

virtual void stopInteractionGeneration (
    InteractionClassHandle theHandle) // supplied C1
throw (
    InteractionClassNotPublished,
    FederateInternalError) = 0;

/////////////////////////////
// Object Management Services //
/////////////////////////////

// 4.4
virtual void discoverObject (
    ObjectId             theObject,      // supplied C1
    ObjectClassHandle    theObjectClass, // supplied C1
    FederationTime       theTime,        // supplied C1
    const UserSuppliedTag theTag,        // supplied C4
    EventRetractionHandle theHandle)    // supplied C1
throw (
    CouldNotDiscover,
    ObjectClassNotKnown,
    InvalidFederationTime,
    FederateInternalError) = 0;
```

```
// 4.5
virtual void reflectAttributeValues (
    ObjectID                      theObject,      // supplied C1
    const AttributeHandleValuePairSet& theAttributes, // supplied C4
    FederationTime                 theTime,        // supplied C1
    const UserSuppliedTag          theTag,         // supplied C4
    EventRetractionHandle          theHandle)     // supplied C1
throw (
    ObjectNotKnown,
    AttributeNotKnown,
    InvalidFederationTime,
    FederateInternalError) = 0;

// 4.7
virtual void receiveInteraction (
    InteractionClassHandle         theInteraction, // supplied C1
    const ParameterHandleValuePairSet& theParameters, // supplied C4
    FederationTime                 theTime,        // supplied C1
    const UserSuppliedTag          theTag,         // supplied C4
    EventRetractionHandle          theHandle)     // supplied C1
throw (
    InteractionClassNotKnown,
    InteractionParameterNotKnown,
    InvalidFederationTime,
    FederateInternalError) = 0;

// 4.9
virtual void removeObject (
    ObjectID                      theObject,      // supplied C1
    ObjectRemovalReason            theReason,      // supplied C1
    FederationTime                 theTime,        // supplied C1
    const UserSuppliedTag          theTag,         // supplied C4
    EventRetractionHandle          theHandle)     // supplied C1
throw (
    ObjectNotKnown,
    InvalidFederationTime,
    FederateInternalError) = 0;

virtual void removeObject (
    ObjectID                      theObject,      // supplied C1
    ObjectRemovalReason            theReason)     // supplied C1
    FederationTime                 theTime,        // supplied C1
    const UserSuppliedTag          theTag)         // supplied C4
throw (
    ObjectNotKnown,
    InvalidFederationTime,
    FederateInternalError) = 0;

// 4.15
virtual void provideAttributeValueUpdate (
    ObjectID                      theObject,      // supplied C1
    const AttributeHandleSet& theAttributes) // supplied C4
throw (
    ObjectNotKnown,
```

```
AttributeNotKnown,
FederateInternalError) = 0;

// 4.17
virtual void reflectRetraction (
    EventRetractionHandle theHandle) // supplied C1
throw (
    EventNotKnown,
    FederateInternalError) = 0;

///////////////////////////////
// Ownership Management Services //
///////////////////////////////

// 5.2
virtual AttributeHandleSet&                               // returned C6
requestAttributeOwnershipAssumption (
    ObjectId          theObject,           // supplied C1
    const AttributeHandleSet& offeredAttributes, // supplied C4
    const UserSuppliedTag   theTag)           // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotKnown,
    AttributeAlreadyOwned,
    FederateInternalError) = 0;

// 5.3
virtual void attributeOwnershipDivestitureNotification (
    ObjectId          theObject,           // supplied C1
    const AttributeHandleSet& releasedAttributes) // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError) = 0;

// 5.4
virtual void attributeOwnershipAcquisitionNotification (
    ObjectId          theObject,           // supplied C1
    const AttributeHandleSet& securedAttributes) // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError) = 0;

// 5.6
virtual AttributeHandleSet&                               // returned C6
requestAttributeOwnershipRelease (
    ObjectId          theObject,           // supplied C1
    const AttributeHandleSet& candidateAttributes, // supplied C4
    const UserSuppliedTag   theTag)           // supplied C4
throw (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError) = 0;
```

```
// 5.8
virtual void informAttributeOwnership (
    ObjectID      theObject,      // supplied C1
    AttributeHandle theAttribute, // supplied C1
    FederateHandle theOwner)     // supplied C1
throw (
    ObjectNotKnown,
    AttributeNotKnown,
    FederateInternalError) = 0;

///////////////////////////////
// Time Management Services //
///////////////////////////////

// 6.10
virtual void timeAdvanceGrant (
    FederationTime theTime) // supplied C1
throw (
    InvalidFederationTime,
    TimeAdvanceWasNotInProgress,
    FederationTimeAlreadyPassed,
    FederateInternalError) = 0;

/////////////////////////////
// Data Distribution Management //
/////////////////////////////

// 7.4
virtual void changeThresholds (
    Region      theRegion,      // supplied
    ThresholdSet& theThresholds) // returned
throw (
    RegionNotKnown,
    FederateInternalError) = 0;
```

```
//File: RTI.hh

#ifndef RTI_hh
#define RTI_hh

#include <fstream.h>
#include <math.h>

struct RTIambPrivate;

class RTI {
public:

    #include "baseTypes.hh"
    #include "RTITypes.hh"

    class RTIambassador {
public:
    #include "RTIambServices.hh"
private:
    RTIambPrivate* privateData;
};

    class FederateAmbassador {
public:
    #include "federateAmbServices.hh"
};
};

#endif
```

10 HLA Ada 95 Application Programmer's Interface

```
--  
-- File: rti.1.adb  
--  
  
with System;  
with RTI_Base_Types;  
with Unchecked_Conversion;  
  
-- with RTI_Basic_Lists;  
package RTI is  
  
--  
-- system parameters  
--  
-- These are from the C++. Probably all are not needed, but they are  
-- all reproduced here. Easier to delete than to add.  
--  
    Max_Federation : constant Positive := 20;  
    Max_Federate : constant Positive := 20;  
    Max_Object_Classes : constant Positive := 100;  
    Max_Interaction_Classes : constant Positive := 100;  
    Max_Attributes_Per_Class : constant Positive := 50;  
    Max_Parameters_Per_Class : constant Positive := 50;  
    Max_Bytes_Per_Value : constant Positive := 256;  
    Max_Federation_Name_Length : constant Positive := 64;  
    Max_Federate_Name_Length : constant Positive := 64;  
    Privilege_To_Delete_Name : constant String := "privilegeToDelete";  
    Privilege_To_Delete_Handle : constant := 0;  
    Root_Object_Class_Handle : constant := 0;  
    Root_Interaction_Class_Handle : constant := 0;  
  
--  
-- types used across the RTI  
--  
    type Simple_Handle is new RTI_Base_Types.RTI_UShort;  
  
    type Value_Ptr is access System.Address;  
    type Value_Bucket is record  
        The_Length: positive;  
        The_Value: Value_Ptr;  
    end record;  
  
    type Federation_Execution_Name is new RTI_Base_Types.RTI_String;  
    type Federation_Handle is new Simple_Handle;  
  
    type Federation_Time is new RTI_Base_Types.RTI_Duration;  
    type Tick_Time is new RTI_Base_Types.RTI_Duration;
```

```
type Federate_Name is new RTI_Base_Types.RTI_String;
type Federate_Handle is new Simple_Handle;

type Object_Id is new RTI_Base_Types.RTI_ULong;

type Object_Class_Name is new RTI_Base_Types.RTI_String;
type Object_Class_Handle is new Simple_Handle;

type Interaction_Class_Name is new RTI_Base_Types.RTI_String;
type Interaction_Class_Handle is new Simple_Handle;

type Attribute_Name is new RTI_Base_Types.RTI_String;

type Attribute_Handle is new Simple_Handle;

-- 
-- List Types And Declarations
--

type Node_Item;
type Node_Ptr is access all Node_Item'Class;
type Node_Item is tagged
  record
    Next : Node_Ptr := null;
  end Record;

type Attribute_Handle_Set_Element is new Node_Item
with record
  The_Attribute_Handle : Attribute_Handle;
end record;
type Attribute_Handle_Set is new Node_Ptr;

type Attribute_Handle_Value_Pair_Set_Element is
  new Node_Item
with record
  The_Attribute_Handle: Attribute_Handle;
  The_Attribute_Value: Value_Bucket;
end record;
type Attribute_Handle_Value_Pair_Set is new Node_Ptr;

type Parameter_Handle_Value_Pair_Set_Element is
  new Node_Item
with record
  The_Parameter_Handle: Attribute_Handle;
  The_Parameter_Value: Value_Bucket;
end record;
type Parameter_Handle_Value_Pair_Set is new Node_Ptr;

-- 
-- End List Types
-- 

type Parameter_Name is new RTI_Base_Types.RTI_String;
```

```
type Parameter_Handle is new Simple_Handle;

type Pause_Label is new RTI_Base_Types.RTI_String;

type Save_Label is new RTI_Base_Types.RTI_String;

type Event_Retraction_Handle is private;

-- type User_Supplied_Tag is new Value_Bucket; -- WHY?????
type User_Supplied_Tag is new RTI_Base_Types.RTI_String;

type RTI_Boolean is (RTI_False, RTI_True);
for RTI_Boolean use
    (RTI_False => 0,
     RTI_True => 1);

procedure Set_Retraction_Handle -- New
    (Handle : in out Event_Retraction_Handle;
     Value : in Simple_Handle);

function Convert_To_Integer is new Unchecked_Conversion
    (Source => RTI_Boolean,
     Target => Integer);

function Convert_To_Rti_Boolean is new Unchecked_Conversion
    (Source => Integer,
     Target => RTI_Boolean);

private

type Event_Retraction_Handle is new Simple_Handle;

end RTI;
```

```
--  
-- File: rti_base_types.1.ada  
--  
  
with Interfaces.C;  
  
package RTI_Base_Types is  
  
    package C renames Interfaces.C;  
  
    --  
    -- base types  
    --  
    subtype RTI_String is String;  
  
        -- RTI_Float isn't used, is it needed?  
    --    subtype RTI_Float is Float;  
  
    subtype RTI_Duration is Duration;  
  
    subtype RTI_ULong is integer range 0 .. Integer'Last;  
    subtype RTI_Long is integer;  
  
    subtype RTI_Ushort is C.Unsigned_Short;  
    subtype RTI_Short is C.Short;  
    subtype RTI_Double is C.Double;  
    subtype RTI_Float is C.C_Float;  
    subtype RTI_Id is C.Unsigned_Long;  
    -- subtype RTI_Boolean is C.Unsigned_Char; -- Huh????  
  
end RTI_Base_Types;
```

```
--  
-- File: rti_basic_lists.1.adb  
--  
-----  
-- | NAME rti_basic_lists.1.adb  
-- | PURPOSE  
-- |  
-- | ENVIRONMENT_DEPENDENCIES None  
-- | MACHINE_DEPENDENCIES designed for use on IBM PowerPC running AIX  
-- | DEVICE_IO_DEPENDENCIES None  
-- | EXECUTION_CONSTRAINTS None  
-- | DESIGN_CONSTRAINTS designed to function with version F0 of the RTI  
-- |  
-----  
  
with RTI;  
  
package RTI_Basic_Lists is  
  
    -- type Node_Item is tagged limited private;  
    -- type Node_Ptr is access all Node_Item'Class;  
  
    -- abstraction assumes head of a list is a Node_Ptr  
  
    function Size_Of (The_List : in RTI.Node_Ptr) return Natural;  
  
    procedure Empty (The_List : in out RTI.Node_Ptr);  
  
    procedure Add (The_List : in out RTI.Node_Ptr;  
                  The_Item : in RTI.Node_Item'Class);  
  
    procedure Remove (The_List : in out RTI.Node_Ptr;  
                      The_Item : in RTI.Node_Item'Class);  
  
    function Is_Empty (The_List : in RTI.Node_Ptr) return Boolean;  
  
    function Get (The_List : in RTI.Node_Ptr;  
                 Ith_Item: Positive) return RTI.Node_Item'Class;  
  
generic  
    with function Equal (Item_A: in RTI.Node_Item'Class;  
                        Item_B: in RTI.Node_Item'Class) return Boolean;  
    function Is_Member (The_List : in RTI.Node_Item'Class;  
                       The_Item : in RTI.Node_Item'Class) return Boolean;  
  
end RTI_Basic_Lists;
```

```
--  
-- File: rti_ambassador_class.1.ada  
--  
  
with Federate_Ambassador_Class;  
  
--  
-- RTI_Ambassador_Class represents the class of objects (and services  
-- associated with them) which provide the interface to the RTI. A new RTI  
-- Ambassador Object becomes associated with each new "join" by a simulation  
-- to the Federation Execution  
--  
  
package RTI_Ambassador_Class is  
  
    -- The type from which RTI Ambassadors will be instantiated  
    -- This tagged type contains a reference to the associated Federate  
    -- Ambassador object. This is needed due to the way Ada95 associates  
    -- routines with objects. This differs from the C++ paradigm in which  
    -- routines are tightly bound to the object definition  
  
    type RTI_Ambassador is tagged  
        record  
            This_Federate_Ambassador :  
                Federate_Ambassador_Class.Federate_Ambassador_Ptr;  
        end record;  
  
    type RTI_Ambassador_Ptr is access RTI_Ambassador;  
  
end RTI_Ambassador_Class;
```

```
--  
-- File: federate_ambassador_class.1.adb  
--  
  
-----  
-- | NAME federate_ambassador_class.1.adb  
-- | PURPOSE  
-- |   Provide an abstract class definition for the Federate_Ambassador Object.  
-- |   Federates need to "inherit" from the template and define the member  
-- |   functions listed here. Instantiations of this class will be called  
-- |   by the RTI  
-- |  
-- | ENVIRONMENT_DEPENDENCIES None  
-- | MACHINE_DEPENDENCIES designed for use on IBM PowerPC running AIX  
-- | DEVICE_IO_DEPENDENCIES None  
-- | EXECUTION_CONSTRAINTS None  
-- | DESIGN_CONSTRAINTS designed to function with version F0 of the RTI  
-- |  
-----  
  
--  
-- Federate_Ambassador_Class provides a "template" for federate supplied  
-- routines. Each of the federate supplied routines are initiated by the  
-- RTI via callbacks  
--  
  
with RTI;  
with DDM;  
  
package Federate_Ambassador_Class is  
  
  -- Federate_Ambassador is an abstract tagged type because it is up to the  
  -- the individual Federates to instantiate this object along with its  
  -- corresponding routines.  
  type Federate_Ambassador is abstract tagged  
    record  
      Identifier : Integer;  
    end record;  
  
  --  
  -- Federate_Ambassador_Ptr provides access to the class wide set of  
  -- Federate Ambassadors. This allows dispatching to the appropriate  
  -- routine as additional Federate Ambassadors are created.  
  type Federate_Ambassador_Ptr is access Federate_Ambassador'Class;  
  
  --  
  --  
  -- Provides the Federation Management services (2.x services) initiated  
  -- by the RTI.  
  --  
  
  -- 2.6  
  procedure Initiate_Pause  
    (The_Federate_Ambassador : in Federate_Ambassador;
```

```
The_Label : in RTI.Pause_Label) is abstract;
-- should raise only the following exceptions:
--     Federate_Already_Paused
--     Federate_Internal_Error

-- 2.9
procedure Initiate_Resume
    (The_Federate_Ambassador : in Federate_Ambassador) is abstract;
-- should raise only the following exceptions:
--     Federate_Not_Paused
--     Federate_Internal_Error

-- 2.12
procedure Initiate_Federate_Save
    (The_Federate_Ambassador : in Federate_Ambassador;
     The_Label : in RTI.Save_Label) is abstract;
-- should raise only the following exceptions:
--     Unable_To_Perform_Save
--     Federate_Internal_Error

procedure Initiate_Federate_Save
    (The_Federate_Ambassador : in Federate_Ambassador;
     The_Label : in RTI.Save_Label;
     The_Time : in RTI.Federation_Time) is abstract;
-- should raise only the following exceptions:
--     Invalid_Federation_Time
--     Unable_To_Perform_Save
--     Federate_Internal_Error

-- 2.16
procedure Initiate_Restore
    (The_Federate_Ambassador : in Federate_Ambassador;
     The_Label : in RTI.Save_Label) is abstract;
-- should raise only the following exceptions:
--     Specified_Save_Label_Does_Not_Exist
--     Could_Not_Restore
--     Federate_Internal_Error

--
--
-- Provides the Declaration Management services (3.x services) initiated
-- by the RTI.
--

-- 3.5
procedure Start_Updates
    (The_Federate_Ambassador : in Federate_Ambassador;
     The_Class : in RTI.Object_Class_Handle;
     The_Attributes : access RTI.Attribute_Handle_Set) is abstract;
-- should raise only the following exceptions:
--     Object_Class_Not_Published
--     Attribute_Not_Published
--     Federate_Internal_Error
```

```
procedure Stop_Updates
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Class : in RTI.Object_Class_Handle;
   The_Attributes : access RTI.Attribute_Handle_Set) is abstract;
-- should raise only the following exceptions:
--   Object_Class_Not_Published
--   Attribute_Not_Published
--   Federate_Internal_Error

-- 3.6
procedure Start_Interaction_Generation
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Handle : RTI.Interaction_Class_Handle) is abstract;
-- should raise only the following exceptions:
--   Interaction_Class_Not_Published
--   Federate_Internal_Error

procedure Stop_Interaction_Generation
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Handle : in RTI.Interaction_Class_Handle) is abstract;
-- should raise only the following exceptions:
--   Interaction_Class_Not_Published
--   Federate_Internal_Error

--
--
-- Provides the Object Management services (4.x services) initiated
-- by the RTI.
--

-- types unique to Internal Object Management services

type Object_Removal_Reason is (Out_Of_Region,
                                 Object_Deleted,
                                 No_Longer_Subscribed);

for Object_Removal_Reason use
  (Out_Of_Region => 0,
   Object_Deleted => 1,
   No_Longer_Subscribed => 2);

-- 4.4
procedure Discover_Object
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   The_Object_Class : in RTI.Object_Class_Handle;
   The_Time : in RTI.Federation_Time;
   The_Tag : in RTI.User_Supplied_Tag;
   The_Event : in RTI.Event_Retraction_Handle) is abstract;
-- should raise only the following exceptions:
--   Could_Not_Discover_Object
--   Object_Class_Not_Known
--   Invalid_Federation_Time
```

```
-- Federate_Internal_Error

-- 4.5
procedure Reflect_Attribute_Values
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   The_Attributes : access RTI.Attribute_Handle_Value_Pair_Set;
   The_Time : in RTI.Federation_Time;
   The_Tag : in RTI.User_Supplied_Tag;
   The_Event : in RTI.Event_Retraction_Handle) is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Invalid_Federation_Time
--   Federate_Internal_Error

-- 4.7
procedure Receive_Interaction
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Interaction_Class : in RTI.Interaction_Class_Handle;
   The_Parameters : access RTI.Parameter_Handle_Value_Pair_Set;
   The_Time : in RTI.Federation_Time;
   The_Tag : in RTI.User_Supplied_Tag;
   The_Event : in RTI.Event_Retraction_Handle) is abstract;
-- should raise only the following exceptions:
--   Interaction_Class_Not_Known
--   Interaction_Parameter_Not_Known
--   Invalid_Federation_Time
--   Federate_Internal_Error

-- 4.9
procedure Remove_Object
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   The_Reason : in Object_Removal_Reason;
   The_Time : in RTI.Federation_Time;
   The_Tag : in RTI.User_Supplied_Tag;
   The_Event : in RTI.Event_Retraction_Handle) is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Invalid_Federation_Time
--   Federate_Internal_Error

procedure Remove_Object
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   The_Reason : in Object_Removal_Reason;
   The_Time : in RTI.Federation_Time;
   The_Tag : in RTI.User_Supplied_Tag) is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Invalid_Federation_Time
--   Federate_Internal_Error
```

```
-- 4.15
procedure Provide_Attribute_Value_Update
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   The_Attributes : access RTI.Attribute_Handle_Set) is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Federate_Internal_Error

-- 4.17
procedure Reflect_Retraction
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Event : in RTI.Event_Retraction_Handle) is abstract;
-- should raise only the following exceptions:
--   Event_Not_Known
--   Federate_Internal_Error

--
--
-- Provides the Ownership Management services (5.x services) initiated
-- by the RTI.
--

-- 5.2
function Request_Attribute_Ownership_Assumption
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   Offered_Attributes : access RTI.Attribute_Handle_Set;
   The_Tag : in RTI.User_Supplied_Tag)
return RTI.Attribute_Handle_Set is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Attribute_Already_Owned
--   Federate_Internal_Error

function Request_Attribute_Ownership_Assumption
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   Offered_Attributes : access RTI.Attribute_Handle_Set)
return RTI.Attribute_Handle_Set is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Federate_Internal_Error

-- 5.3
procedure Attribute_Ownership_Divestiture_Notification
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
```

```
Released_Attributes : access RTI.Attribute_Handle_Set) is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Federate_Internal_Error

-- 5.4
procedure Attribute_Ownership_Acquisition_Notification
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   Acquired_Attributes : access RTI.Attribute_Handle_Set) is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Federate_Internal_Error

-- 5.6
function Request_Attribute_Ownership_Release
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object : in RTI.Object_Id;
   Requested_Attributes : access RTI.Attribute_Handle_Set;
   The_Tag : in RTI.User_Supplied_Tag)
return RTI.Attribute_Handle_Set is abstract;
-- should raise only the following exceptions:
--   Object_Not_Known
--   Attribute_Not_Known
--   Federate_Internal_Error

-- 5.7
procedure Inform_Attribute_Ownership
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Object: in RTI.Object_Id;
   The_Attribute: in RTI.Attribute_Handle;
   The_Owner: in RTI.Federate_Handle) is abstract;
-- should raise only the following exceptions:
--   Federate_Internal_Error

--
--
-- Provides the Time Management services (6.x services) initiated
-- by the RTI.
--

-- 6.10
procedure Time_Advance_Grant
  (The_Federate_Ambassador : in Federate_Ambassador;
   The_Time : in RTI.Federation_Time) is abstract;
-- should raise only the following exceptions:
--   Invalid_Federation_Time
--   Time_Advance_Was_Not_In_Progress
--   Federation_Time_Already_Passed
--   Federate_Internal_Error
```

```
--  
--  
-- Provides the Data Distribution Management services (7.x services) initiated  
-- by the RTI.  
--  
-- 7.4  
procedure Change_Thresholds  
  (The_Federate_Ambassador : in Federate_Ambassador;  
   The_Region : in DDM.Region;  
   The_Thresholds : access DDM.Threshold_Set) is abstract;  
-- should raise only the following exceptions:  
--   Region_Not_Known  
--   Federate_Internal_Error  
  
end Federate_Ambassador_Class;
```

```
--  
-- File rac_federation_management.1.adb  
--  
  
with RTI;  
with RTI_Base_Types;  
  
package RTI_Ambassador_Class.Federation_Management is  
  
--  
-- Provides the Federation Management services (2.x services) initiated  
-- by federates.  
--  
  
--  
-- types unique to Federation Management services  
--  
  
    type File_Name is new RTI_Base_Types.RTI_String;  
  
    type Resign_Action is (Release_Attributes,  
                           Delete_Objects,  
                           Delete_Objects_And_Release_Attributes);  
  
    for Resign_Action use  
        (Release_Attributes => 1,  
         Delete_Objects => 2,  
         Delete_Objects_And_Release_Attributes => 3);  
  
    type Save_Indicator is (Save_Achieved,  
                           Save_Not_Achieved);  
  
    type Restore_Indicator is (Restore_Federate_Achieved,  
                               Restore_Federate_Not_Achieved);  
  
--  
-- 2.x services  
--  
  
-- 2.1  
procedure Create_Federation_Execution  
    (The_Name : in RTI.Federation_Execution_Name;  
     R_I_D : in File_Name);  
-- should raise ONLY the following exceptions:  
--     Federation_Execution_Already_Exists  
--     Could_Not_Open RID  
--     Error_Reading RID  
--     RTI_Internal_Error  
  
-- 2.2  
procedure Destroy_Federation_Execution  
    (The_RTI_Ambassador : in RTI_Ambassador_Ptr;  
     Execution_Name : in RTI.Federation_Execution_Name);
```

```
-- should raise ONLY the following exceptions:  
--   Federates_Currently_Joined  
--   Federation_Execution_Does_Not_Exist  
--   RTI_Internal_Error  
  
-- 2.3  
procedure Join_Federation_Execution  
  (Your_Name : in RTI.Federate_Name;  
   Execution_Name : in RTI.Federation_Execution_Name;  
   The_RTI_Ambassador : in RTI_Ambassador_Ptr);  
-- should raise ONLY the following exceptions:  
--   Federate_Already_Execution_Member  
--   Federation_Execution_Does_Not_Exist  
--   Could_Not_Open RID  
--   Error_Reading RID  
--   RTI_Internal_Error  
  
-- 2.4  
procedure Resign_Federation_Execution  
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;  
   The_Action : in Resign_Action);  
-- should raise ONLY the following exceptions:  
--   Federate_Owns_Attributes  
--   Federate_Not_Execution_Member  
--   Federation_Execution_Does_Not_Exist  
--   Invalid_Resign_Action  
--   RTI_Internal_Error  
  
-- not implemented in F.0  
procedure Resign_Federation_Execution  
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;  
   A_Federate : in RTI.Federate_Handle;  
   The_Action : in Resign_Action);  
-- should raise ONLY the following exceptions:  
--   Federate_Owns_Attributes  
--   Federation_Execution_Does_Not_Exist  
--   Federate_Not_Execution_Member  
--   RTI_Internal_Error  
  
-- 2.5  
-- not implemented in F.0  
procedure Request_Pause  
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;  
   The_Label : in RTI.Pause_Label);  
-- should raise ONLY the following exceptions:  
--   Federation_Already_Paused  
--   Federate_Not_Execution_Member  
--   RTI_Internal_Error  
  
-- 2.7
```

```
-- not implemented in F.0
procedure Pause_Achieved
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Label : in RTI.Pause_Label);
-- should raise ONLY the following exceptions:
--   Unknown_Label
--   No_Pause_Requested
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 2.8
-- not implemented in F.0
procedure Request_Resume
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr);
-- should raise ONLY the following exceptions:
--   Federation_Not_Paused
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 2.10
-- not implemented in F.0
procedure Resume_Achieved
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr);
-- should raise ONLY the following exceptions:
--   No_Resume_Requested
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 2.11
-- not implemented in F.0
procedure Request_Federation_Save
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Label : in RTI.Save_Label);
-- should raise ONLY the following exceptions:
--   Save_In_Progress
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- not implemented in F.0
procedure Request_Federation_Save
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Label : in RTI.Save_Label;
   The_Time : in RTI.Federation_Time);
-- should raise ONLY the following exceptions:
--   Save_In_Progress
--   Federation_Time_Already_Passed
--   Invalid_Federation_Time
--   Federate_Not_Execution_Member
--   RTI_Internal_Error
```

```
-- 2.13          -- not implemented in F.0
procedure Federate_Save_Begun
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr);
-- should raise ONLY the following exceptions:
--   Save_Not_Initiated
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

          -- not implemented in F.0
procedure Federate_Save_Begun
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Time : in RTI.Federation_Time);
-- should raise ONLY the following exceptions:
--   Save_Not_Initiated
--   Invalid_Federate_Time
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 2.14          -- not implemented in F.0
procedure Federate_Save_Achieved
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr);
-- should raise ONLY the following exceptions:
--   Save_Not_Initiated
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

          -- not implemented in F.0
procedure Federate_Save_Not_Achieved
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr);
-- should raise ONLY the following exceptions:
--   Save_Not_Initiated
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 2.15          -- not implemented in F.0
procedure Request_Restore
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Label : in RTI.Save_Label);
-- should raise ONLY the following exceptions:
--   Specified_Save_Label_Does_Not_Exist
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 2.17          -- not implemented in F.0
procedure Restore_Achieved
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Label : in RTI.Save_Label);
-- should raise ONLY the following exceptions:
```

```
-- Unknown_Label
-- Restore_Not_Requested
-- Federate_Not_Execution_Member
-- RTI_Internal_Error

-- not implemented in F.0
procedure Restore_Not_Achieved
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Label : in RTI_Save_Label);
-- should raise ONLY the following exceptions:
-- Unknown_Label
-- Restore_Not_Requested
-- Federate_Not_Execution_Member
-- RTI_Internal_Error

end RTI_Ambassador_Class.Federation_Management;
```

```
--  
-- File rac_declaration_management.1.adb  
--  
  
with RTI;  
with DDM;  
  
package RTI_Ambassador_Class.Declaration_Management is  
  
--  
-- Provides the Declaration Management services (3.x services) initiated  
-- by federates.  
--  
  
-- 3.1  
procedure Publish_Object_Class  
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
     The_Class : in RTI.Object_Class_Handle;  
     Attribute_List : access RTI.Attribute_Handle_Set);  
-- should raise ONLY the following exceptions:  
--     Object_Class_Not_Defined  
--     Attribute_Not_Defined  
--     Federate_Not_Execution_Member  
--     RTI_Internal_Error  
  
procedure Unpublish_Object_Class  
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
     The_Class : in RTI.Object_Class_Handle);  
-- should raise ONLY the following exceptions:  
--     Object_Class_Not_Defined  
--     Federate_Owes_Attributes  
--     Federate_Not_Execution_Member  
--     RTI_Internal_Error  
  
-- 3.2  
procedure Publish_Interaction_Class  
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
     The_Interaction : in RTI.Interaction_Class_Handle);  
-- should raise ONLY the following exceptions:  
--     Interaction_Class_Not_Defined  
--     Federate_Not_Execution_Member  
--     RTI_Internal_Error  
  
procedure Unpublish_Interaction_Class  
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
     The_Interaction : in RTI.Interaction_Class_Handle);  
-- should raise ONLY the following exceptions:  
--     Interaction_Class_Not_Defined  
--     Federate_Not_Execution_Member  
--     RTI_Internal_Error
```

```
-- 3.3
procedure Subscribe_Object_Class_Attribute
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Object_Class_Handle;
     Attribute_List : access RTI.Attribute_Handle_Set);
-- should raise ONLY the following exceptions:
--     Object_Class_Not_Defined,
--     Attribute_Not_Defined
--     Federate_Not_Execution_Member
--     RTI_Internal_Error

                                -- not implemented in F.0
procedure Subscribe_Object_Class_Attribute
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Object_Class_Handle;
     The_Attribute : in RTI.Attribute_Handle;
     The_Region : in DDM.Region);
-- should raise ONLY the following exceptions:
--     Object_Class_Not_Defined
--     Attribute_Not_Defined
--     Region_Not_Known
--     Federate_Not_Execution_Member
--     RTI_Internal_Error

procedure Unsubscribe_Object_Class_Attribute
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Object_Class_Handle);
-- should raise ONLY the following exceptions:
--     Object_Class_Not_Defined
--     Federate_Not_Execution_Member
--     RTI_Internal_Error

                                -- not implemented in F.0
procedure Unsubscribe_Object_Class_Attribute
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Object_Class_Handle;
     The_Region : in DDM.Region);
-- should raise ONLY the following exceptions:
--     Object_Class_Not_Defined
--     Region_Not_Known
--     Federate_Not_Execution_Member
--     RTI_Internal_Error

-- 3.4
procedure Subscribe_Interaction_Class
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Interaction_Class_Handle);
-- should raise ONLY the following exceptions:
--     Interaction_Class_Not_Defined
```

```
--      Federate_Not_Execution_Member
--      RTI_Internal_Error

                           -- not implemented in F.0
procedure Subscribe_Interaction_Class
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Interaction_Class_Handle;
     The_Region : in DDM.Region);
-- should raise ONLY the following exceptions:
--      Interaction_Class_Not_Defined
--      Region_Not_Known
--      Federate_Not_Execution_Member
--      RTI_Internal_Error

procedure Unsubscribe_Interaction_Class
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Interaction_Class_Handle);
-- should raise ONLY the following exceptions:
--      Interaction_Class_Not_Defined
--      Federate_Not_Execution_Member
--      RTI_Internal_Error

                           -- not implemented in F.0
procedure Unsubscribe_Interaction_Class
    (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
     The_Class : in RTI.Interaction_Class_Handle;
     The_Region : in DDM.Region);
-- should raise ONLY the following exceptions:
--      Interaction_Class_Not_Defined
--      Region_Not_Known
--      Federate_Not_Execution_Member
--      RTI_Internal_Error

end RTI_Ambassador_Class.Declaration_Management;
```

```
--  
-- File rac_object_management.1.ada  
--  
  
with Rti;  
with Rti_Basic_Lists;  
  
package Rti_Ambassador_Class.Object_Management is  
  
    --  
    -- provides the object management services (2.x services) initiated  
    -- by federates.  
    --  
  
    --  
    -- types unique to object management services  
    --  
  
    subtype Object_Id_Count is Positive;  
  
    type Data_Order_Type is (Receive,  
                           Timestamp);  
    for Data_Order_Type use  
        (Receive => 1,  
         Timestamp => 2);  
  
    type Transportation_Type is (Reliable,  
                                 Best_Effort);  
    for Transportation_Type use  
        (Reliable => 1,  
         Best_Effort => 2);  
  
    -- 4.1  
    procedure Request_Id  
        (The_Rti_Ambassador : in Rti_Ambassador_Ptr;  
         Id_Count : in Object_Id_Count;  
         First_Id: Rti.Object_Id;  
         Last_Id: Rti.Object_Id);  
    -- should raise only the following exceptions:  
    --      too_many_ids_requested  
    --      id_supply_exhausted  
    --      federate_not_execution_member  
    --      rti_internal_error  
  
    -- 4.2  
    procedure Register_Object  
        (The_Rti_Ambassador : in Rti_Ambassador_Ptr;  
         The_Object_Class : in Rti.Object_Class_Handle;  
         The_Object : in Rti.Object_Id);  
    -- should raise only the following exceptions:  
    --      invalid_object_id  
    --      object_id_already_registered  
    --      object_class_not_defined
```

```
--      object_class_not_published
--      federate_not_execution_member
--      rti_internal_error

-- 4.3
function Update_Attribute_Values
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Object : in Rti.Object_Id;
   The_Attributes_And_Values : access Rti.Attribute_Handle_Value_Pair_Set;
   The_Time : in Rti.Federation_Time;
   The_Tag : in Rti.User_Supplied_Tag)
return Rti.Event_Retraction_Handle;
-- should raise only the following exceptions:
--      object_not_know
--      attribute_not_defined
--      attribute_not_owned
--      invalid_federation_time
--      federate_not_execution_member
--      rti_internal_error

-- 4.6
function Send_Interaction
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Interaction : in Rti.Interaction_Class_Handle;
   The_Parameters_And_Values : access Rti.Parameter_Handle_Value_Pair_Set;
   The_Time : in Rti.Federation_Time;
   The_Tag : in Rti.User_Supplied_Tag)
return Rti.Event_Retraction_Handle;
-- should raise only the following exceptions:
--      interaction_class_not_defined
--      interaction_class_not_published
--      interaction_parameter_not_defined
--      invalid_federation_time
--      federate_not_execution_member
--      rti_internal_error

-- 4.8
function Delete_Object
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Object : in Rti.Object_Id;
   The_Time : in Rti.Federation_Time;
   The_Tag : in Rti.User_Supplied_Tag)
return Rti.Event_Retraction_Handle;
-- should raise only the following exceptions:
--      object_not_known
--      delete_privilege_not_held
--      invalid_federation_time
--      federate_not_execution_member
--      rti_internal_error

-- 4.10
procedure Change_Attribute_Transportation_Type
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
```

```
The_Object : in Rti.Object_Id;
The_Attributes : access Rti.Attribute_Handle_Set;
The_Type : in Transportation_Type);
-- should raise only the following exceptions:
--   object_not_known
--   attribute_not_defined
--   attribute_not_owned
--   invalid_transportation_type
--   federate_not_execution_member
--   rti_internal_error

-- 4.11
procedure Change_Attribute_Order_Type
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Object : in Rti.Object_Id;
   The_Attributes : access Rti.Attribute_Handle_Set;
   The_Data_Order : in Data_Order_Type);
-- should raise only the following exceptions:
--   object_not_known
--   attribute_not_defined
--   attribute_not_owned
--   invalid_order_type
--   federate_not_execution_member
--   rti_internal_error

-- 4.12
procedure Change_Interaction_Transportation_Type
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Interaction_Class : in Rti.Interaction_Class_Handle;
   The_Type : in Transportation_Type);
-- should raise only the following exceptions:
--   interaction_class_not_defined
--   federate_not_publishing_interaction_class
--   invalid_transportation_type
--   federate_not_execution_member
--   rti_internal_error

-- 4.13
procedure Change_Interaction_Order_Type
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Interaction_Class : in Rti.Interaction_Class_Handle;
   The_Data_Order : in Data_Order_Type);
-- should raise only the following exceptions:
--   interaction_class_not_defined
--   federate_not_publishing_interaction_class
--   invalid_order_type
--   federate_not_execution_member
--   rti_internal_error

-- 4.14
procedure Request_Object_Attribute_Value_Update
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Object : in Rti.Object_Id;
```

```
The_Attributes : access Rti.Attribute_Handle_Set);
-- should raise only the following exceptions:
--   object_not_known
--   attribute_not_defined
--   federate_not_execution_member
--   rti_internal_error

procedure Request_Object_Attribute_Value_Update
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Object : in Rti.Object_Id);
-- should raise only the following exceptions:
--   object_not_known
--   federate_not_execution_member
--   rti_internal_error

procedure Request_Class_Attribute_Value_Update
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Class : in Rti.Object_Class_Handle;
   The_Attributes : access Rti.Attribute_Handle_Set);
-- should raise only the following exceptions:
--   object_class_not_defined
--   attribute_not_defined
--   federate_not_execution_member
--   rti_internal_error

procedure Request_Class_Attribute_Value_Update
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Class : in Rti.Object_Class_Handle);
-- should raise only the following exceptions:
--   object_class_not_defined
--   federate_not_execution_member
--   rti_internal_error

-- 4.16
procedure Retract
  (The_Rti_Ambassador : in Rti_Ambassador_Ptr;
   The_Event : in Rti.Event_Retraction_Handle);
-- should raise only the following exceptions:
--   invalid_retraction_handle
--   federate_not_execution_member
--   rti_internal_error

end Rti_Ambassador_Class.Object_Management;
```

```
--  
-- File rac_ownership_management.1.ada  
--  
  
with RTI_Base_Types;  
-- with RTI_Basic_Lists;  
with RTI;  
  
package RTI_Ambassador_Class.Ownership_Management is  
  
--  
-- Provides the Ownership Management services (5.x services) initiated  
-- by federates.  
--  
  
--  
-- types unique to Ownership Management services  
--  
  
    type Ownership_Divestiture_Condition is (Negotiated,  
                                              Unconditional);  
  
    for Ownership_Divestiture_Condition use  
        (Negotiated => 1,  
         Unconditional => 2);  
  
    type Federate_Set_Element is new RTI.Node_Item with record  
        The_Federate: RTI.Federate_Handle;  
    end record;  
    type Federate_Handle_Set is new RTI.Node_Ptr;  
  
    -- 5.1  
    procedure Request_Attribute_Ownership_Divestiture  
        (The_RTI_Ambassador :in RTI_Ambassador_Ptr;  
         The_Object : in RTI.Object_Id;  
         The_Attributes : access RTI.Attribute_Handle_Set;  
         The_Condition : in Ownership_Divestiture_Condition;  
         The_Tag : in RTI.User_Supplied_Tag);  
    -- should raise ONLY the following exceptions:  
    --     Object_Not_Known  
    --     Attribute_Not_Defined  
    --     Attribute_Not_Owned  
    --     Invalid_Divestiture_Condition  
    --     Federate_Not_Execution_Member  
    --     RTI_Internal_Error  
  
    procedure Request_Attribute_Ownership_Divestiture  
        (The_RTI_Ambassador :in RTI_Ambassador_Ptr;  
         The_Object : in RTI.Object_Id;  
         The_Attributes : access RTI.Attribute_Handle_Set;  
         The_Condition : in Ownership_Divestiture_Condition;  
         The_Tag : in RTI.User_Supplied_Tag;  
         The_Candidates : access Federate_Handle_Set);
```

```
-- should raise ONLY the following exceptions:  
--   Object_Not_Known  
--   Attribute_Not_Defined  
--   Attribute_Not_Owned  
--   Invalid_Divestiture_Condition  
--   Federate_Does_Not_Exist  
--   Federate_Not_Execution_Member  
--   RTI_Internal_Error  
  
-- 5.5  
procedure Request_Attribute_Ownership_Acquisition  
  (The_RTI_Ambassador :in RTI_Ambassador_Ptr;  
   The_Object : in RTI.Object_Id;  
   Desired_Attributes : access RTI.Attribute_Handle_Set;  
   The_Tag : in RTI.User_Supplied_Tag);  
-- should raise ONLY the following exceptions:  
--   Object_Not_Known  
--   Object_Class_Not_Published  
--   Object_Class_Not_Subscribed  
--   Attribute_Not_Defined  
--   Attribute_Not_Published  
--   Attribute_Not_Subscribed  
--   Federate_Owes_Attributes  
--   Federate_Not_Execution_Member  
--   RTI_Internal_Error  
  
-- 5.7  
procedure Query_Attribute_Ownership  
  (The_RTI_Ambassador :in RTI_Ambassador_Ptr;  
   The_Object : in RTI.Object_Id;  
   The_Attribute : in RTI.Attribute_Handle);  
-- should raise ONLY the following exceptions:  
--   Object_Not_Known  
--   Attribute_Not_Defined  
--   Attribute_Not_Known  
--   Federate_Not_Execution_Member  
--   RTI_Internal_Error  
  
function Attribute_Is_Owned_By_Federate  
  (The_RTI_Ambassador :in RTI_Ambassador_Ptr;  
   The_Object : in RTI.Object_Id;  
   The_Attribute : in RTI.Attribute_Handle)  
return RTI.RTI_Boolean;  
-- should raise ONLY the following exceptions:  
--   Object_Not_Known  
--   Attribute_Not_Defined  
--   Attribute_Not_Known  
--   Federate_Not_Execution_Member  
--   RTI_Internal_Error  
  
end RTI_Ambassador_Class.Ownership_Management;
```



```
--  
-- File rac_time_management.1.ada  
--  
  
with RTI;  
  
package RTI_Ambassador_Class.Time_Management is  
  
--  
-- Provides the Time Management services (6.x services) initiated  
-- by federates.  
--  
  
--  
-- types unique to Time Management services  
--  
  
    subtype Federation_Time_Delta is RTI.Federation_Time;  
  
    -- 6.1  
    function Request_Federation_Time  
        (The_RTI_Ambassador : in RTI_Ambassador_Ptr)  
    return RTI.Federation_Time;  
    -- should raise ONLY the following exceptions:  
    --      Federate_Not_Execution_Member  
    --      RTI_Internal_Error  
  
    -- 6.2  
    function Request_Lower_Bound_Timestamp  
        (The_RTI_Ambassador : in RTI_Ambassador_Ptr)  
    return RTI.Federation_Time;  
    -- should raise ONLY the following exceptions:  
    --      Federate_Not_Execution_Member  
    --      RTI_Internal_Error  
  
    -- 6.3  
    function Request_Federate_Time  
        (The_RTI_Ambassador : in RTI_Ambassador_Ptr)  
    return RTI.Federation_Time;  
    -- should raise ONLY the following exceptions:  
    --      Federate_Not_Execution_Member  
    --      RTI_Internal_Error  
  
    -- 6.4  
    function Request_Minimum_Next_Event_Time  
        (The_RTI_Ambassador : in RTI_Ambassador_Ptr)  
    return RTI.Federation_Time;  
    -- should raise ONLY the following exceptions:  
    --      Federate_Not_Execution_Member  
    --      RTI_Internal_Error;
```

```
-- 6.5
procedure Set_Lookahead
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Lookahead : in Federation_Time_Delta);
-- should raise ONLY the following exceptions:
--   Invalid_Federation_Time_Delta
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 6.6
function Request_Lookahead
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr)
return Federation_Time_Delta;
-- should raise ONLY the following exceptions:
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 6.7
procedure Time_Advance_Request
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Time : in RTI.Federation_Time);
-- should raise ONLY the following exceptions:
--   Time_Advance_Already_In_Progress
--   Federation_Time_Already_Passed
--   Invalid_Federation_Time
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 6.8
procedure Next_Event_Request
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Time : in RTI.Federation_Time);
-- should raise ONLY the following exceptions:
--   Time_Advance_Already_In_Progress
--   Federation_Time_Already_Passed
--   Invalid_Federation_Time
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- not implemented in F.0
procedure Next_Event_Request
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Time : in RTI.Federation_Time;
   Delta_Time : in RTI.Federation_Time);
-- should raise ONLY the following exceptions:
--   Time_Advance_Already_In_Progress
--   Federation_Time_Already_Passed
--   Invalid_Federation_Time
--   Federate_Not_Execution_Member
--   RTI_Internal_Error
```

```
-- 6.9
procedure Flush_Queue_Request
  (The_RTI_Ambassador : in RTI_Ambassador_Ptr;
   The_Time : in RTI.Federation_Time);
-- should raise ONLY the following exceptions:
--   Time_Advance_Already_In_Progress
--   Federation_Time_Already_Passed
--   Invalid_Federation_Time
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

end RTI_Ambassador_Class.Time_Management;
```

```
--  
-- File: ddm.1.ada  
--  
  
with RTI_Base_Types;  
-- with RTI_Basic_Lists;  
with RTI;  
  
package DDM is  
  
    type Space_Name is new RTI_BASE_TYPES.RTI_String;  
    type Space_Handle is private;  
  
    type Region is private;  
  
    type RTI_Range is record  
        first, last: RTI_Base_Types.RTI_ULong;  
    end record;  
  
    type Extents_Element is new RTI.Node_Item with record  
        The_Range: RTI_Range;  
    end record;  
    type Extents is new RTI.Node_Ptr;  
  
    type Extent_Set_Element is new RTI.Node_Item with record  
        The_Extent: Extents;  
    end record;  
    type Extent_Set is new RTI.Node_Ptr;  
  
    type Threshold is new RTI_Base_Types.RTI_ULong;  
  
    type Threshold_Set_Element is new RTI.Node_Item with record  
        The_Threshold: Threshold;  
    end record;  
    type Threshold_Set is new RTI.Node_Ptr;  
  
    type Association_Action is (Form_Association,  
                                Break_Association);  
  
    procedure Set_Region  
        (My_Region : in out Region;  
         With_Value : in RTI_Base_Types.RTI_Ulong);  
  
private  
  
    type Region is new RTI_Base_Types.RTI_ULong;  
  
    type Space_Handle is new RTI.Simple_Handle;  
  
end DDM;
```

```
--  
-- File rac_data_distribution_management.1.ada  
--  
  
with RTI_BASE_TYPES;  
with RTI;  
with DDM;  
  
package RTI_Ambassador_Class.Data_Distribution_Management is  
  
--  
-- Provides the Data Distribution Management services  
-- (7.x services) initiated by federates.  
--  
  
-- 7.1  
procedure Create_Update_Region  
(The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
The_Routing_Space : in DDM.Space_Handle;  
The_Extents : access DDM.Extent_Set;  
The_Region : out DDM.Region);  
  
-- should raise ONLY the following exceptions:  
-- Space_Not_Defined  
-- Invalid_Extents  
-- Federate_Not_Execution_Member  
-- RTI_Internal_Error  
  
-- 7.2  
procedure Create_Subscription_Region  
(The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
The_Routing_Space : in DDM.Space_Name;  
The_Extents : access DDM.Extent_Set;  
The_Region : out DDM.Region);  
-- should raise ONLY the following exceptions:  
-- Space_Not_Defined  
-- Invalid_Extents  
-- Federate_Not_Execution_Member  
-- RTI_Internal_Error  
  
-- 7.3  
procedure Associate_Update_Region  
(The_RTI_Ambassador: in RTI_Ambassador_Ptr;  
The_Region : in DDM.Region;  
The_Object : in RTI.Object_Id;  
The_Attributes : access RTI.Attribute_Handle_Set;  
The_Action : in DDM.Association_Action);  
-- should raise ONLY the following exceptions:  
-- Region_Not_Known  
-- Object_Not_Known  
-- Attribute_Not_Defined  
-- Federate_Not_Execution_Member  
-- RTI_Internal_Error
```

```
procedure Associate_Update_Region
  (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
   The_Region : in DDM.Region;
   The_Interaction_Class : in RTI.Interaction_Class_Handle;
   The_Action : in DDM.Association_Action);
-- should raise ONLY the following exceptions:
--   Region_Not_Known
--   Interaction_Class_Not_Known
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 7.5
procedure Modify_Region
  (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
   The_Region : in DDM.Region;
   The_Extents : access DDM.Extent_Set);
-- should raise ONLY the following exceptions:
--   Region_Not_Known
--   Invalid_Extents
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 7.6
procedure Delete_Region
  (The_RTI_Ambassador: in RTI_Ambassador_Ptr;
   The_Region : in DDM.Region);
-- should raise ONLY the following exceptions:
--   Region_Not_Known
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 
-- 8.9
function Get_Space_Handle
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_Name:
     in DDM.Space_Name)
return DDM.Space_Handle;
-- should raise only the following exceptions:
--   Name_Not_Found
--   Federate_Not_Execution_Member
--   Concurrent_Access_Attempted
--   RTI_Internal_Error
--   Unimplemented_Service

-- 8.10
function Get_Space_Name
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_Handle: DDM.Space_Handle)
```

```
    return DDM.Space_Name;
-- should raise only the following exceptions:
--   Space_Not_Defined
--   Federate_Not_Execution_Member
--   Concurrent_Access_Attempted
--   RTI_Internal_Error
--   Unimplemented_Service

end RTI_Ambassador_Class.Data_Distribution_Management;
```

```
--  
-- File: rti_support.1.adb  
--  
  
with RTI;  
with RTI_Base_Types;  
with RTI_Ambassador_Class.Data_Distribution_Management;  
  
package RTI_Support is  
  
    -- 8.1  
    function Get_Object_Class_Handle  
        (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;  
         The_Name: in RTI.Object_Class_Name)  
        return RTI.Object_Class_Handle;  
    -- should raise only the following exceptions:  
    --     Name_Not_Found  
    --     Federate_Not_Execution_Member  
    --     Concurrent_Access_Attempted  
    --     RTI_Internal_Error  
  
    -- 8.2  
    function Get_Object_Class_Name  
        (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;  
         The_Handle: in RTI.Object_Class_Handle)  
        return RTI.Object_Class_Name;  
    -- should raise only the following exceptions:  
    --     Object_Class_Not_Defined  
    --     Federate_Not_Execution_Member  
    --     Concurrent_Access_Attempted  
    --     RTI_Internal_Error  
  
    -- 8.3  
    function Get_Attribute_Handle  
        (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;  
         The_Name: in RTI.Attribute_Name;  
         Which_Class: in RTI.Object_Class_Handle)  
        return RTI.Attribute_Handle;  
    -- should raise only the following exceptions:  
    --     Object_Class_Not_Defined  
    --     Name_Not_Found  
    --     Federate_Not_ExecutionMember  
    --     Concurrent_Access_Attempted  
    --     RTI_Internal_Error  
  
    -- 8.4  
    function Get_Attribute_Name  
        (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;  
         The_Handle: in RTI.Attribute_Handle;  
         Which_Class: in RTI.Object_Class_Handle)  
        return RTI.Attribute_Name;  
    -- should raise only the following exceptions:
```

```
-- Object_Class_Not_Defined
-- Attribute_Not_Defined
-- Federate_Not_Execution_Member
-- Concurrent_Access_Attempted
-- RTI_Internal_Error;

-- 8.5
function Get_Interaction_Class_Handle
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_Name: in RTI.Interaction_Class_Name)
  return RTI.Interaction_Class_Handle;
-- should raise only the following exceptions:
--   Name_Not_Found
--   Federate_Not_Execution_Member
--   Concurrent_Access_Attempted
--   RTI_Internal_Error

-- 8.6
function Get_Interaction_Class_Name
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_Handle: in RTI.Interaction_Class_Handle)
  return RTI.Interaction_Class_Name;
-- should raise only the following exceptions:
--   Interaction_Class_Not_Defined
--   Federate_Not_Execution_Member
--   Concurrent_Access_Attempted
--   RTI_Internal_Error

-- 8.7
function Get_Parameter_Handle
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_Name: in RTI.Parameter_Name;
   Which_Class: in RTI.Interaction_Class_Handle)
  return RTI.Parameter_Handle;
-- should raise only the following exceptions:
--   Interaction_Class_Not_Defined
--   Name_Not_Found
--   Federate_Not_Execution_Member
--   Concurrent_Access_Attempted
--   RTI_Internal_Error

-- 8.8
function Get_Parameter_Name
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_Handle: in RTI.Parameter_Handle;
   Which_Class: in RTI.Interaction_Class_Handle)
  return RTI.Parameter_Name;
-- should raise only the following exceptions:
--   Interaction_Class_Not_Defined
--   Interaction_Parameter_Not_Defined
--   Federate_Not_Execution_Member
```

```
-- Concurrent_Access_Attempted
-- RTI_Internal_Error

-- 8.13
procedure Turn_Regulation_On
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr);
-- should raise only the following exceptions:
--   Federation_Time_Already_Passed
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

function Turn_Regulation_On_Now
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr)
return RTI.Federation_Time;
-- should raise only the following exceptions:
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

procedure Turn_Regulation_Off
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr);
-- should raise only the following exceptions:
--   Federate_Not_Execution_Member
--   RTI_Internal_Error

-- 8.14
procedure Set_Time_Constrained
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   The_State: RTI.RTI_Boolean);
-- should raise only the following exceptions:
--   Federate_Not_Execution_Member
--   RTI_Internal_Error;

-- 8.15
function Tick
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr)
return RTI.RTI_Boolean;
-- should raise only the following exceptions:
--   RTI_Internal_Error;

function Tick
  (The_RTI_Ambassador: in RTI_Ambassador_Class.RTI_Ambassador_Ptr;
   Minimum : in RTI.Tick_Time;
   Maximum : in RTI.Tick_Time)
return RTI.RTI_Boolean;
-- should raise only the following exceptions:
--   RTI_Internal_Error;

end RTI_Support;
```

11 References

[HLA DEF] “Preliminary Definition, High Level Architecture,” briefing, Defense Modeling and Simulation Office, latest version on WEB.

[HLA OMT] “High Level Architecture Object Model Template”, Defense Modeling and Simulation Office, latest version on WEB.

[HLA DDM] “HLA Data Distribution Management Design Document”, Defense Modeling and Simulation Office, version release TBD.

[HLA TM] “HLA Time Management: Design Document”, Defense Modeling and Simulation Office, latest version on WEB.

COMMENTS

Comments on this document should be sent by electronic mail to the Defense Modeling and Simulation Office (DMSO) HLA Specifications e-mail address (hla_specs@msis.dmso.mil). The subject line of the message should include the section and line numbers referenced in the comment. The body of each submittal should include: (1) the name and E-mail address of the person making the comment (separate from the mail header), (2) reference to the RTI service or portion of the I/F Specification that the comment addresses (by section number, page, and line number), (3) a one sentence summary of the comment/issue, (4) a brief description of the comment/issue, and (5) any suggested resolution or action to be taken.